



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.1 CHYBY TISKU

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

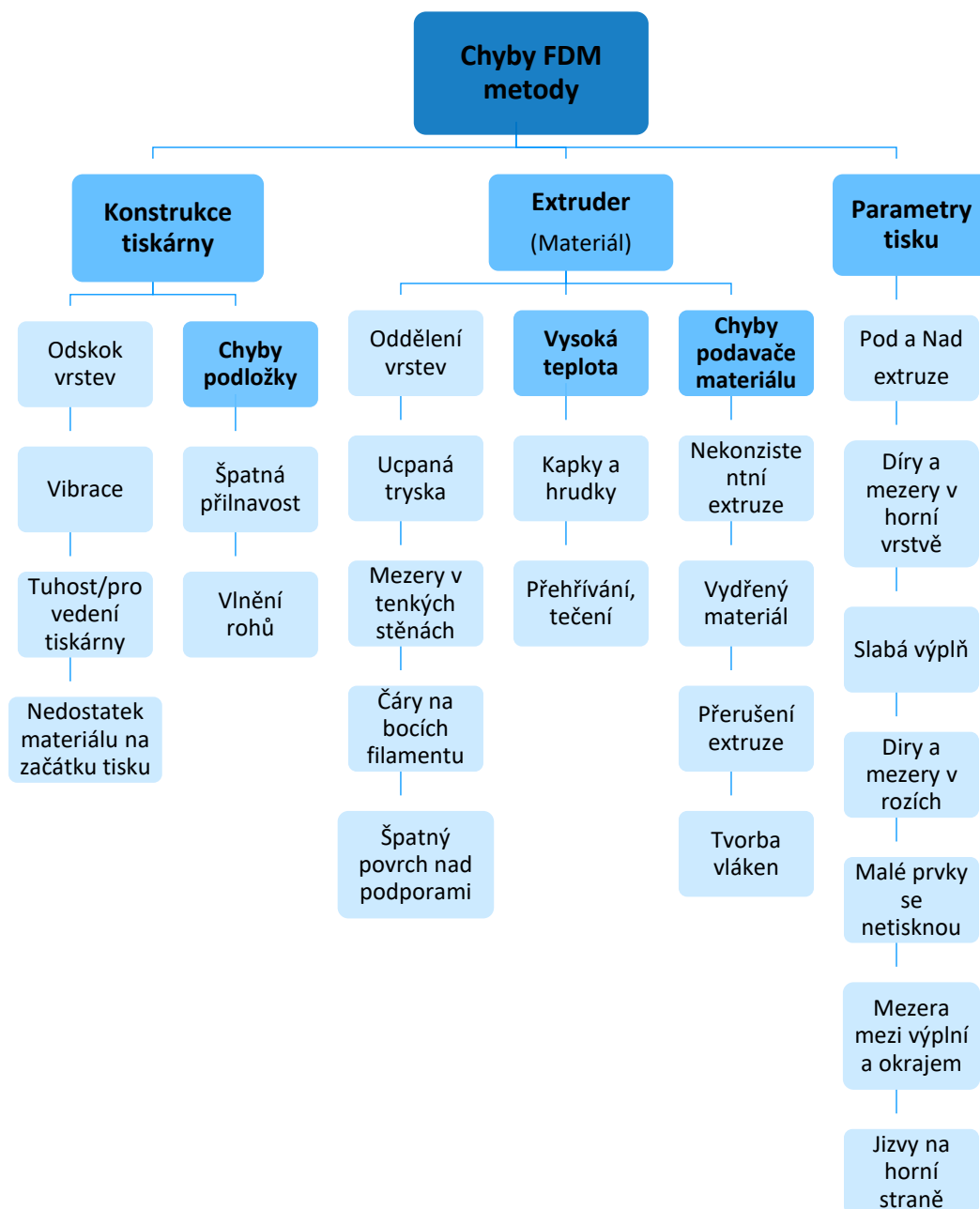
SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# 1 CHYBY VZNIKAJÍCÍ PŘI FDM TISKU

Následující kapitola je zaměřena na přehled chyb vzniklých u 3D tisku metodou Fused Deposition Modeling. Tyto chyby nejčastěji záleží na vlastnostech tiskárny, materiálu a parametrů tisku [3]. Nastavením vhodných parametrů a vhodným seřízením tiskárny se lze v mnoha případech těmto chybám vyhnout.



obr. 1-1 Schéma chyb FDM tisku [3]

## 1.1 Přehled nejčastějších chyb FDM tisku

### Nedostatek materiálu na začátku tisku

Při novém tisku se z trysky vytlačuje nejen materiál, ale také částečně vzduch [3]. To vede ke špatnému začátku tisku. Může dojít k tomu, že první vrstva nebude ideálně spojena, což má za následek zrušení tisku.

#### Příčiny:

- a) Vytečení materiálu z trysky po předchozím tisku.
- b) Velká mezera při tisku první vrstvy.

#### Řešení:

- a) Přidání funkce skirt, která vytiskne část materiálu před samotným tiskem, odstranění vzduchu z trysky.
- b) Kalibrace osy Z.

### Špatná přilnavost k podložce

Důležité pro tisk nového dílu je, aby první vrstva dokonale přilnula k podložce [3]. V opačném případě by mohlo dojít k pozdějšímu odtržení od podložky.

#### Příčiny:

- a) Podložka není ve vodorovné poloze. Mezera mezi podložkou a tryskou je rozdílná.
- b) Tryska je příliš vzdálená od podložky.
- c) Vysoká rychlost tisku první vrstvy.
- d) Špatná teplota materiálu a podložky (teploty pro různé materiály se liší/špatně definovaný materiál v softwaru).
- e) Povrch podložky.

#### Řešení:

- a) Vyrovnání podložky nebo kalibrace osy Z.
- b) Kalibrace osy Z a koncového snímače nebo lze vyřešit softwarově - nastavením počáteční polohy trysky.
- c) Snížení rychlosti tisku první vrstvy například o polovinu.
- d) Nastavení správné teploty pro daný materiál.
- e) Použití lepidla, papírové pásy, odmaštění povrchu.

Pokud nic z předchozích řešení nepomůže, tiskárny disponují funkcemi brim a raft, které rozšíří první vrstvy do stran [3]. Dojde ke zvýšení plochy, kterou drží model na podložce.

### **Pod-extruze**

Tryska nevytlačuje dostatek materiálu, který je potřebný ke spojení jednotlivých vrstev [3]. Vznikají mezery mezi tištěnými vrstvami.

#### **Příčiny:**

- a) Špatný průměr vlákna materiálu.
- b) Snížené vytlačování materiálu.

#### **Řešení:**

- a) Nastavení vhodného průměru vlákna v softwaru, většinou bývá 1,75 mm nebo 2,85 mm.
- b) Nevhodné nastavení funkce flow rate, která umožňuje regulaci vytlačovaného materiálu.

### **Nad-extruze**

Nad extruze je děj, při kterém tryška vytlačuje příliš materiálu [3]. Je tedy opakem pod extruze.

### **Přehřívání, tečení**

Při tisku je materiál zahřát nad teplotu skelného přechodu materiálu, je tedy zaručené vytlačování skrz tryšku [3]. Poté materiál tuhne a drží svůj tvar. Musí být tedy zachována rovnováha mezi teplotou roztaveného materiálu a rychlostí ochlazování. Jinak dochází k problémům s kvalitou tisku.

#### **Příčiny:**

- a) Nedostatečné chlazení materiálu.
- b) Příliš vysoká teplota materiálu při závěrečných vrstvách.
- c) Vysoká rychlost tisku.

#### **Řešení:**

- a) Změna otáček chladicího ventilátoru.
- b) Softwarově nastavení nižší teploty tavení nebo nižší teplota materiálu při tisku závěrečných vrstev.
- c) Snížení rychlosti tisku nebo snížení rychlosti tisku závěrečných vrstev.

Jestliže předchozí metody nepomůžou, je možné tisknout dvě tělesa zároveň. Přidáním druhého tělesa dojde k navýšení celkového času, a tedy se prodlouží doba chlazení materiálu tištěného modelu.

## **Odskok vrstev**

Tento problém nastává v okamžiku, kdy se aktuální pozice trysky liší od pozice v g-codu [3]. Dojde tedy k posunutí vrstvy a celý tisk se posune do stran. Nastává například při přeskočení řemenic. Tato chyba nastává u nízkonákladových tiskáren, bez zpětnovazebního řízení.

### **Příčiny:**

- a) Mechanické problémy-například špatně dotažené řemeny nebo řemenice.
- b) Elektrické problémy-například poškozené kabely.

### **Řešení:**

- a) Kontrola mechanických částí tiskárny.
- b) Výměna elektrických kabelů.

## **Oddělování vrstev**

Vrstvy jsou tištěny jedna na druhou, dokud se vrstvy dostatečně nespojí [3]. Výsledný model se může rozdělit.

### **Příčiny:**

- a) Výška vrstvy je příliš vysoká.
- b) Nízká teplota materiálu, některé materiály jsou náchylné na teplotu tisku.

### **Řešení:**

- a) Nevhodné nastavení kroku vůči průměru trysky, snížení výšky vrstvy.
- b) Zvýšení teploty tavení materiálu.

## **Rozměrová přesnost**

Vytištěný model se neshoduje s navrhnutým 3D modelem [3].

### **Příčiny:**

- a) Tisk první vrstvy, může ovlivnit 10 až 20 následujících vrstev.
- b) Chyba softwarového rozměru.
- c) Zmenšování rozměrů modelu důsledkem teplotní roztažnosti.

### **Řešení:**

- a) Kalibrace osy Z, nastavení správné výšky první vrstvy a její pozici v ose Z.
- b) Kompenzace horizontální velikosti v softwaru.
- c) Zvětšení modelu pomocí měřítka.



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.2 DEWE-50-USB2-8 DATASHEET

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# USB Devices with 24 bit Resolution

- Isolated DAQP modules
- Easy to install on your computer
- 24 bit resolution
- 204.8 kS/s per channel, simultaneous sampling
- Synchronous CAN interfaces and counter/digital inputs
- DEWESoft-7-SE and OPT-CAN included



Add your choice of signal conditioning,  
A/D board(s) and software to complete these systems

Specifications	DEWE-50-USB2-8
<b>Analog input</b>	
Number of channels	8 (simultaneously sampled)
Measured values	According installed DAQP modules
<b>Internal A/D system</b>	
Resolution	24 bit
Type of ADC	Sigma-Delta
Sampling rate	204.8 kS/s
-3 dB bandwidth	76 kHz @ 204.8 kS/s (consider possible limit of DAQP module)
Accuracy	±0.1 % of range, ±0.5 mV
Signal to noise @ fs<1000 Hz	< 100 dB
Crosstalk	< 100 dB
<b>Counter/Digital inputs</b>	
Number of channels	2 counters or 6 digital inputs (per software each counter can be selected to be 3x digital input)
Counter modes	Event counting, encoder input, period, pulsewidth, duty cycle, frequency measurement
Resolution	32 bit
Time base	102.4 MHz
Signal levels	TTL/CMOS
Input voltage protection	30 V
<b>CAN inputs</b>	
Number of channels	2
Specification	CAN 2.0B, up to 1 MBaud
Physical layer	High speed
<b>Environmental</b>	
Operating temperature	0 to 50°C
Storage temperature	-20 to 70°C
Relative humidity	95 % non condensing @ 60°C
Vibration	tbd
Shock	tbd
<b>Processing</b>	
System	Requires PC based system with DEWESoft software
Interface	USB 2.0
<b>Power requirements</b>	
Supply voltage (max.)	10 to 36 V <sub>DC</sub>
Typical power consumption	Typ. 20 W (5 W internal A/D system + DAQP modules)
<b>Physical</b>	
Dimensions (L x W x H)	230 x 181 x 104 mm (9.06 x 7.13 x 4.09 in.)
Weight	Typ. 3 kg (2.5 kg + DAQP modules) 6.6 lb. (5.5 lb. + DAQP modules)
<b>Software</b>	
Displays	Recorder, Scope, FFT, 3D Waterfall FFT, Octave, ...
Triggers	Edge, Filtered Edge, Window, Pulsewidth, Slope, FFT, ...
Online standard mathematics	Formula editor, FIR-, IIR-, FFT-filter, basic statistics, reference curve
Online special mathematics	Human Body Vibration, Order Tracking, Rotational & Torsional Vibration, Sound Level, Frequency Response Function
<b>System options</b>	
<b>Option</b>	<b>Description</b>
50-8-OUT-5	8 BNC connectors on back panel, ±5 V output of DAQP-modules
50-8-SYNC	Synchronization option for two DEWE-50-USB2-8. Allows using max. two units as a 16 channel system, synchronization cable 50-8-CBL-SYNC-x needs to be ordered additionally.

NEW  
DEWE2

Dedicated Instruments

Instruments

Front-ends

Signal Conditioning

Components

Software

## Analog Input

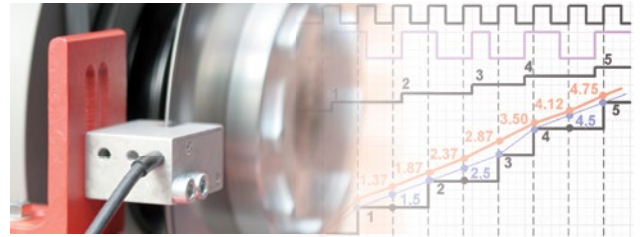
The internal A/D system offers eight analog inputs, each has its own sigma-delta A/D converter and is sampled at up to 204.8 kS/s at 24 bit resolution. Anti-aliasing filters are included for each channel.

The **DEWE-50-USB2-8** offers eight slots for high performance DAQP isolated signal conditioning modules. Thus any analog sensor can be connected.



## Counter/Digital Input

There are Lemo sockets where each can either be used as one counter/encoder input or as three digital inputs – this is a software selection and can be set individually for each socket. Thanks to the special DEWETRON technology, the counter/digital inputs are acquired absolutely synchronously to the analog channels. DEWETRON counters are able to perform



- Basic counting
- Gated counting
- Up/down counting
- Duty cycle
- Frequency measurement
- Pulse width measurement
- Period time measurement
- Two pulse edge separation



## CAN Interface

There are two high speed CAN interfaces which are able to acquire data from vehicle CAN – or vehicle OBDII interface – as well as from any sensor outputting CAN data.

Alternatively DEWETRON CPAD2 modules can be connected to a CAN interface to acquire quasi-static thermocouple, RTD, voltage or current signals.

## SYNC Interface

A SYNC interface enables combination of two units to a 16 channel system.







# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.3 ARDUINO MEGA 2560 DATASHEET

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

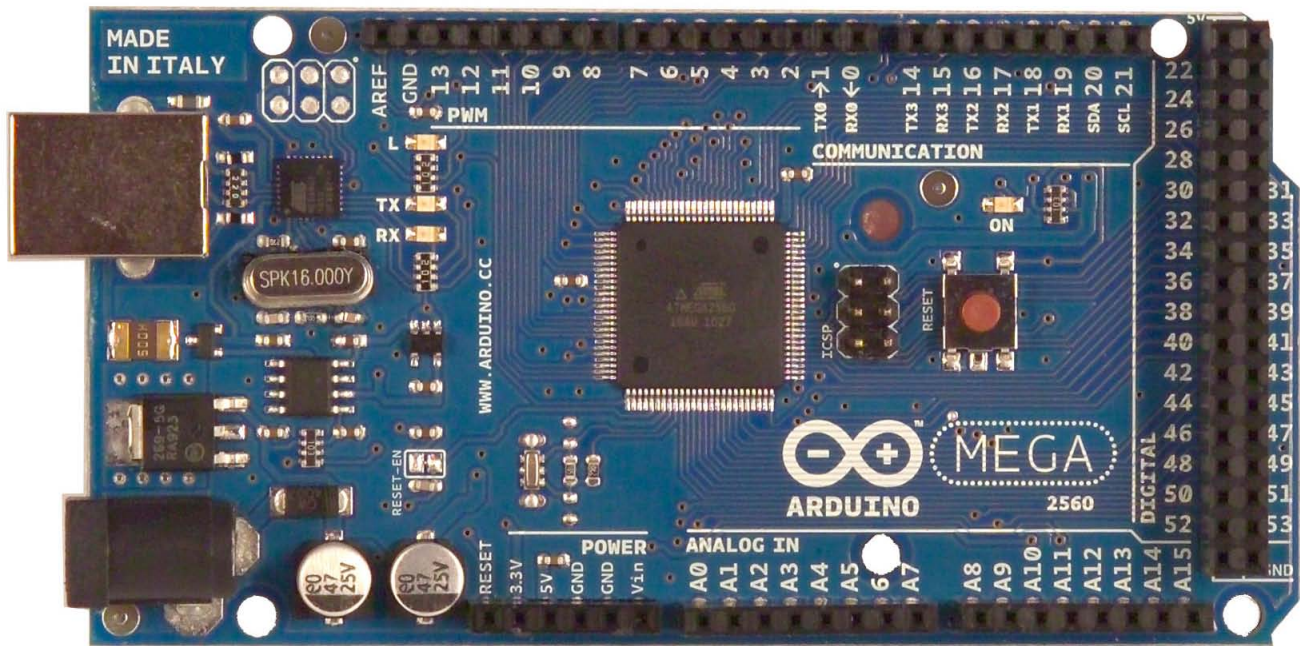
### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# Arduino Mega 2560



The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

## Schematic & Reference Design

EAGLE files: [arduino-mega2560-reference-design.zip](#)

Schematic: [arduino-mega2560-schematic.pdf](#)

## Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

## Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- + **VIN**. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- + **5V**. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- + **3V3**. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- + **GND**. Ground pins.

## Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- + **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX)**. Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- + **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2)**. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- + **PWM: 0 to 13**. Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- + **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS)**. These pins support SPI communication using the [SPI library](#). The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- + **LED: 13**. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- + **I<sup>2</sup>C: 20 (SDA) and 21 (SCL)**. Support I<sup>2</sup>C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I<sup>2</sup>C pins on the Duemilanove or Diecimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

✚ **AREF**. Reference voltage for the analog inputs. Used with [`analogReference\(\)`](#).

✚ **Reset**. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

## Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega2560's digital pins.

The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation on the Wiring website](#) for details. For SPI communication, use the [SPI library](#).

## Programming

The Arduino Mega can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available [in the Arduino repository](#). The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can



have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega2560 contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

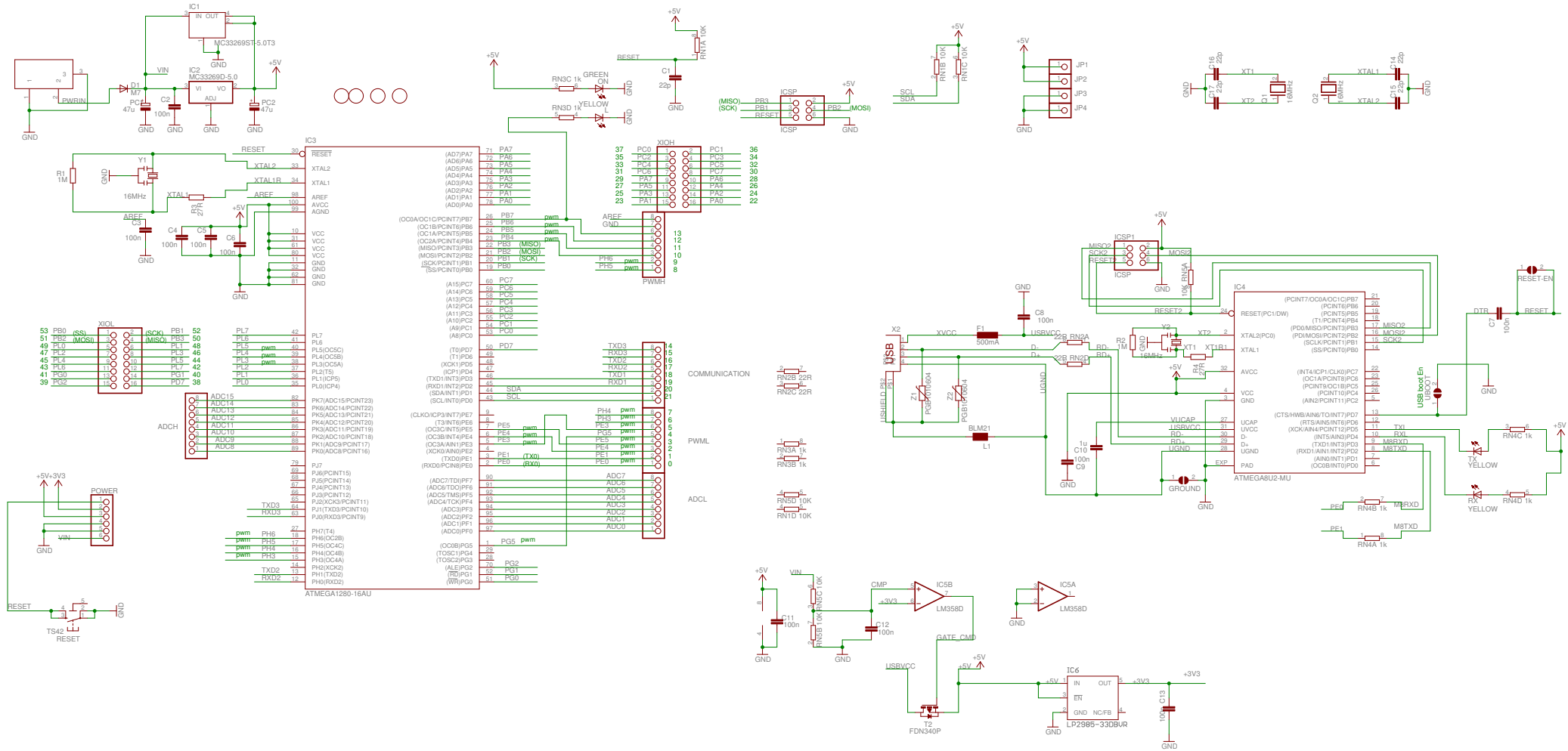
### USB Overcurrent Protection

The Arduino Mega2560 has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

### Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega2560 PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega2560 is designed to be compatible with most shields designed for the Uno, Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega2560 and Duemilanove / Diecimila. *Please note that  $I^2C$  is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).*





# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.4 BAUMER CH8501 OADM 20I4460 S14C DATASHEET

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

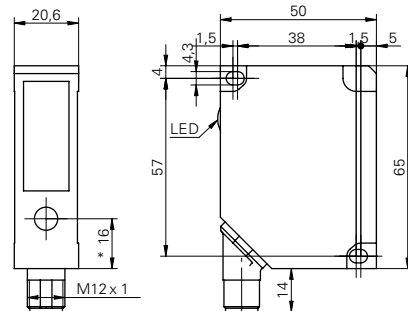
**BRNO 2019**



## Distance sensors

## OADM 20I4460/S14C

## dimension drawing



\* emitter axis

## general data

measuring distance Sd	30 ... 130 mm
adjustment	no
power on indication	LED green
soiled lens indicator	LED red
resolution	0,05 ... 0,07 mm
linearity error	$\pm 0,15 \dots \pm 0,22$ mm
light source	pulsed red laser diode
wave length	650 nm
laser class	2
beam type	point
beam diameter	2 ... 1 mm
temperature drift	< 0,03 % Sde/K

## electrical data

response time / release time	< 10 ms
voltage supply range +Vs	12 ... 28 VDC
current consumption max.	100 mA
output circuit	analog
output signal	4 ... 20 mA / 0 ... 10 VDC
load resistance (analog I)	< (+Vs - 6 V) / 0,02 A
load resistance (analog U)	> 100 kOhm
output current	< 100 mA
alarm output	PNP
short circuit protection	yes
reverse polarity protection	yes, Vs to GND

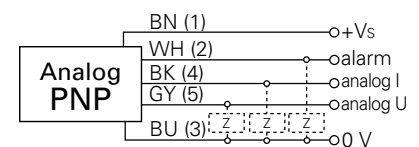
## mechanical data

width / diameter	20,6 mm
height / length	65 mm
depth	50 mm
type	rectangular
housing material	die-cast zinc
front (optics)	glass
connection types	connector M12 5 pin, rotatable

## photo



## connection diagram



## laser warning



**LASER RADIATION**  
DO NOT STARE INTO BEAM  
Wavelength: 620 ... 680nm  
Max. av. Output: < 1mW  
IEC 60825-1, Ed. 2, 2007  
**CLASS 2 LASER PRODUCT**

Complies with 21 CFR 1040.10 and 1040.11 except for deviations pursuant to laser notice No. 50, dated June 24, 2007

**Distance sensors****OADM 20I4460/S14C****ambient conditions**

operating temperature	0 ... +50 °C
protection class	IP 67



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.5 ZAPOJENÍ LASEROVÉHO SKENOVÁNÍ

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

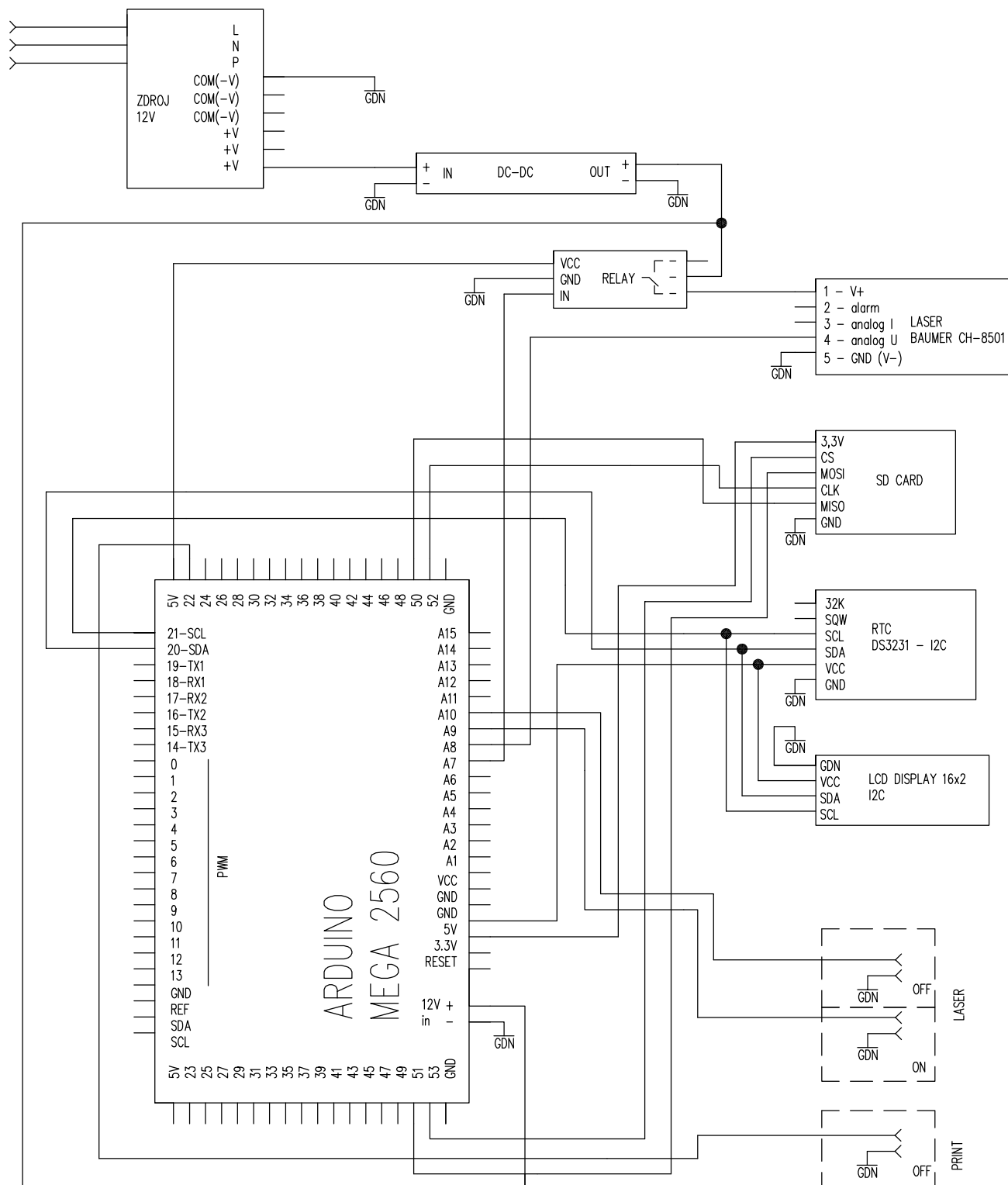
**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE



SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



ZAPOJENÍ LASER  
ZAPOJENÍ KOMUNIKACE

Struktura povrchu:		Hrany:		Měřítko  1:1	Přesnost <b>PŘESNOST</b>
					Tolerování <b>TOLEROVÁNÍ</b>
					Promítání 
Materiál _	Polotovar _			Hmotnost _ kg	CHRÁNĚNO PODLE ISO 16016
  ÚSTAV KONSTRUOVÁNÍ	Druh dokumentu <b>ELEKTRO_SCHÉMA</b>		Název  <b>ZAPOJENÍ_LASER</b>		
	Kreslil <b>RAFAJA</b>				
	Schválil		Číslo dokumentu  <b>0001</b>		
	Datum vydání <b>08.05.2019</b>				
List /					



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.6 G\_CODE

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# 1 G\_CODE

Tato kapitola se zabývá stručným popisem řídicího kódu (G\_codu) pro tiskárnu[35]. Funkce jsou normalizované a používají se pro řízení 3D tiskáren, CNC strojů atd. Pro 3D tiskárnu je řídicí kód generován ve sliceru (Cura, Repetier Host, Grashopper atd.). Pro experimenty byly použity standartní funkce. Pro potřebu dalších experimentů, programování, nastavení byly vytvořeny vlastní funkce. Následuje stručný popis řídicího kódu, upravený pro laserové skenování, automatickou kalibraci atd.

....	Zahájení tisku
....	
G21	Nastaví vstupní jednotky na milimetry
M107	Vypne chlazení ventilátory
G28	Automaticky dojde na nulovou pozici „Homing“
G39	Vyvolání automatické kalibrace (popis viz příloha č. 13)
G92 E0	Nastavení aktuální polohy pro extrudér (E0)
M82	Nastavení absolutního odečítání pro extruder
M106	Zapne chlazení ventilátory
M104 S200	Začne vyhřívat extrudér na teplotu (S200 = 200 °C)
M109 S200	Setrvá na vyhřátí extrudéru na teplotu (S200 = 200 °C)
G1 Z10 F3600	Pohyb na souřadnici „Z“ rychlostí „F“
M140 S80	Začne vyhřívat pracovní desku na teplotu (S80 = 80 °C)
M190 S80	Setrvá na vyhřátí pracovní desky na teplotu (S80 = 80 °C)
G92 E0	Nastavení aktuální polohy pro extrudér (E0)
G1 F1500 E-1	Pohyb na souřadnici „E“ (Extruder) rychlostí „F“
G1 Z0.2 F3600	Pohyb na souřadnici „Z“ rychlostí „F“
....	
....	Tisk vrstvy
....	
G92 E0	Nastavení aktuální polohy pro extrudér (E0)
G28 X0 Y0	Automaticky dojde na nulovou pozici (osa X a Y)
M1 S1	Setrvá (S1) 1 vteřinu (delay pro zahájení komunikace)
M42 P43 S255	Zapne pin 43 - pro zapnutí algoritmu laseru
M1 S1	Setrvá (S1) 1 vteřinu (delay pro zahájení komunikace)
M42 P43 S0	Vypne pin 43

G92 E0	Nastavení aktuální polohy pro extrudér (E0)
G1 F1500 E-1	Pohyb na souřadnici „E“ (Extruder) rychlostí „F“
G1 Z0.2 F3600	Pohyb na souřadnici „Z“ rychlostí „F“
....	
....	Skenování vrstvy
....	
G92 E0	Nastavení aktuální polohy pro extrudér (E0)
G28 X0 Y0	Automaticky dojde na nulovou pozici (osa X a Y)
M1 S1	Setrvá (S1) 1 vteřinu (delay pro zahájení komunikace)
M42 P42 S255	Zapne pin 42 - pro pozastavení/vypnutí algoritmu laseru
M1 S1	Setrvá (S1) 1 vteřinu (delay pro zahájení komunikace)
M42 P42 S0	Vypne pin 42
G92 E0	Nastavení aktuální polohy pro extrudér (E0)
G1 F1500 E-1	Pohyb na souřadnici „E“ (Extruder) rychlostí „F“
G1 Z0.4 F3600	Pohyb na souřadnici „Z“ rychlostí „F“
....	
....	Tisk další vrstvy následuje další skenování...
....	
G92 E0	Nastavení aktuální polohy pro extrudér (E0)
M400	Dokončí všechny pohyby
M107	Vypne chlazení ventilátory
M104 S0	Ochlazuje extrudér na teplotu (S0 = 0 °C)
G28 X0 Y0	Automaticky dojde na nulovou pozici (osa X a Y)
M84	Zamkne krokové motorky do dalšího pohybu
....	
....	Konec tisku

Popis obsahoval začátek a konec G\_codu pro tisk, který byl v průběhu upravený pro laserové skenování dle potřeby a použití.



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.7 LASER SOFTWARE VAR. 1

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



# 1 LASER SOFTWARE VAR. 1

Program slouží k ovládání modulu monitorování pomocí laserového skenování. Převážně pro zapnutí a vypnutí laseru pomocí komunikace, ukládání dat, vyhodnocování dat a zastavení tisku.

Program monitorovacího modulu

```
#define relay A7                //Definice rele
#include <SPI.h>                 //Definice knihovny
#include <SD.h>                  //Definice knihovny
#include <Wire.h>                //Definice knihovny
#include "DS3231.h"             //Definice knihovny
#include <LiquidCrystal_I2C.h>   //Definice knihovny
#include <EEPROM.h>              //Definice knihovny
LiquidCrystal_I2C lcd(0x27,16,2);
RTCLib RTC;
DS3231 Clock;
int Year;
int Month;
int Date;
int Hour;
int Minute;
int Second;
int tempC;
int interval;
int Second_last;
String buffer;
double LZ;
double DO;
int Vyhodnoceni;
int Vrstva = 0;
int i = 0;
```

```

boolean start = true;

File myFile;

File myFile1;

const int chipSelect = 53;

void setup() {

                                // Put your setup code here, to run once:

    Serial.begin(9600);          // Zahájení sériové komunikace
    pinMode(A8,INPUT);           //
    pinMode(A9,INPUT);           //
    pinMode(A10,INPUT);          //
    pinMode(24,OUTPUT);
    pinMode(relay, OUTPUT);      //
    digitalWrite(relay, HIGH);   //

    //Display
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0,0);
    lcd.print("Laser ver 11");
    lcd.setCursor(0,1);
    lcd.print("Var1 ver6");
    Wire.begin(); //for DS3231
    while (!Serial) {
        ;
    }
    delay(2000);
    Serial.print("Initializing SD card...");
    if (!SD.begin(53)) {
        Serial.println("initialization failed!");
    }
    else
    {
        Serial.println("initialization done.");
    }
}

```

```

}

//Čtení hodnot z SD
Serial.println("Čtu vstupní Data");
myFile1 = SD.open("Data.txt", FILE_READ);
if(myFile1){
    while (myFile1.available()){
        buffer = myFile1.readStringUntil('\n');
        LZ = buffer.toDouble();
        buffer = myFile1.readStringUntil('\n');
        DO = buffer.toDouble();
        break;
    }
    myFile1.close();
} else {
    Serial.println("error opening Data.txt");
}

Serial.println("Končím čtení");
Serial.println("Hodnoty jsou");
Serial.println(LZ);
Serial.println(DO);
}

void loop() {
    // Put your main code here, to run repeatedly:

    //Podmínka komunikace
    if (digitalRead(A9) == HIGH){
        //Reset podmínky pro zastavení(po
        vrstvě se obnoví)
        i=0;
        Vrstva++;
        delay(100);
        //Zapnutí Rele/Laseru
        digitalWrite(relay, LOW);
    }
}

```

```

if (digitalRead(A9)== LOW){
  while (true){
    //čas
    DateTime now = RTC.now();
    Year = now.year();
    Month = now.month();
    Date = now.day();
    Hour = now.hour();
    Minute = now.minute();
    Second = now.second();
    tempC = Clock.getTemperature();
    //Zapnutí Rele/Laseru
    digitalWrite(relay, LOW);
    double sensorValue = analogRead(A8);      //
    double voltage = sensorValue * (10 / 1023.0); // Převede analogový vstup (od 0 do
1023) na napětí (0 - 10V)
    double vzdalenost=(voltage*4.8428)+30.16;   // Přepočet napětí na vzdálenost
    //Porovnávání
    double odchylka=0;
    odchylka = LZ - vzdalenost;
    if (odchylka > DO || odchylka < (-DO)){
      Vyhodnoceni = 1;           //Chyba
      write_data(vzdalenost, Vyhodnoceni,Vrstva, i);
      i++;
    }else{
      Vyhodnoceni = 0;           //Normal
      write_data(vzdalenost, Vyhodnoceni, Vrstva, i);
    }
    //Zastavení tisku
    if (i>15){
      delay(10000);
      digitalWrite(24, HIGH);
    }
  }
}

```

```

    }else{
        digitalWrite(24, LOW);
    }

    //write_data(vzdalenost);          //write data

    if (digitalRead(A10) == HIGH){
        digitalWrite(relay, HIGH);
        //start = false;
        break;
    }
}

}

}

}

}

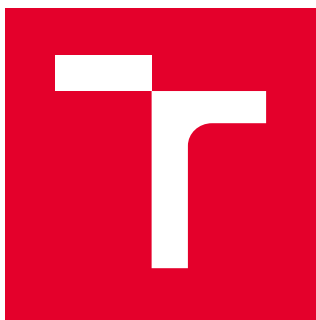
void write_data(double vzdalenost, int Vyhodnoceni, int Vrstva, int i)    //
{
    myFile = SD.open("Laser.csv", FILE_WRITE); /////.txt nebo .csv
    if (myFile) {
        myFile.print(Year);
        myFile.print("/");
        myFile.print(Month);
        myFile.print("/");
        myFile.print(Date);
        myFile.print(";");
        myFile.print(Hour);
        myFile.print(":");
        myFile.print(Minute);
        myFile.print(":");
        myFile.print(Second);
        myFile.print(";");
        myFile.print("Vzdalenost");
        myFile.print(";");
        myFile.print(vzdalenost);
    }
}

```

```
myFile.print(";");
myFile.print("Vyhodnoceni");
myFile.print(";");
myFile.print(Vyhodnoceni);
myFile.print(";");
myFile.print("Vrstva");
myFile.print(";");
myFile.print(Vrstva);
myFile.print(";");
myFile.print("Pocet chyb ve Vrstve");
myFile.print(";");
myFile.print(i);
myFile.print(";");
myFile.println(" ");
myFile.close();
Serial.print(vzdalenost);
Serial.print("    ");
Serial.println("Write file successful!");
} else {
    Serial.println("error opening Laser.csv");
}
}
```

## Program pro zastavení tisku

```
if (start == true){  
  pinMode(6,INPUT);  
  if(digitalRead(6)==HIGH){  
    i++;  
    if(i>=5){  
      for(int f=0;f<6;f++){  
        enqueue_and_echo_commands_P(PSTR("M84 X Y Z E"));  
        SERIAL_ECHOLN("zastavení.....");  
        card.stopSDPrint();  
        safe_delay(1000);  
        watchdog_reset();  
      }  
      safe_delay(1000);  
      enqueue_and_echo_commands_P(PSTR("G72"));  
      watchdog_reset();  
      safe_delay(1000);  
      start=false;  
      watchdog_reset();  
    }  
  }  
}
```



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.8 LASER SOFTWARE VAR. 2

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



# 1 LASER SOFTWARE VAR. 2

Program slouží k ovládání modulu monitorování pomocí laserového skenování. Převážně pro zapnutí a vypnutí laseru pomocí komunikace, ukládání dat.

Program monitorovacího modulu

```
#define relay A7                //Definice rele
#include <SPI.h>                 //Definice knihovny
#include <SD.h>                  //Definice knihovny
#include <Wire.h>                //Definice knihovny
#include "DS3231.h"             //Definice knihovny
#include <LiquidCrystal_I2C.h>   //Definice knihovny
#include <EEPROM.h>              //Definice knihovny

LiquidCrystal_I2C lcd(0x27,16,2);
RTCLib RTC;
DS3231 Clock;

int Year;
int Month;
int Date;
int Hour;
int Minute;
int Second;
int tempC;
int interval;
int Second_last;

String buffer;

double val1;
double val2;

boolean start = true;

File myFile;

File myFile1;

const int chipSelect = 53;
```

```

void setup() {

                                // Put your setup code here, to run once:

    Serial.begin(9600);          // Zahájení sériové komunikace
    pinMode(A8,INPUT);           //
    pinMode(A9,INPUT);           //
    pinMode(A10,INPUT);          //
    pinMode(relay, OUTPUT);       //
    digitalWrite(relay, HIGH);    //

    //Display
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0,0);
    lcd.print("Laser ver 11");
    lcd.setCursor(0,1);
    lcd.print("Var1 ver4");
    Wire.begin(); //for DS3231
    while (!Serial) {
    }
    delay(2000);
    Serial.print("Initializing SD card...");
    if (!SD.begin(53)) {
        Serial.println("initialization failed!");
    }
    else
    {
        Serial.println("initialization done.");
    }

    //Čtení hodnot z SD
    Serial.println("Čtu vstupní Data");
    myFile1 = SD.open("Data.txt", FILE_READ);
    if(myFile1){
        while (myFile1.available()){

```

```

    buffer = myFile1.readStringUntil('\n');
    val1 = buffer.toDouble();
    buffer = myFile1.readStringUntil('\n');
    val2 = buffer.toDouble();
    break;
}
myFile1.close();
}else {
    Serial.println("error opening Data.txt");
}
Serial.println("Končím čtení");
Serial.println("Hodnoty jsou");
Serial.println(val1);
Serial.println(val2);
}

void loop() {
    // Put your main code here, to run repeatedly:
    //Podmínka komunikace
    if (digitalRead(A9) == HIGH){
        while (true){
            //čas
            DateTime now = RTC.now();
            Year = now.year();
            Month = now.month();
            Date = now.day();
            Hour = now.hour();
            Minute = now.minute();
            Second = now.second();
            tempC = Clock.getTemperature();
            //Zapnutí Rele/Laseru
            digitalWrite(relay, LOW);          //
            double sensorValue = analogRead(A8);    //

```

```

        double voltage = sensorValue * (10 / 1023.0); // Převede analogový vstup (od 0 do
1023) na napětí (0 - 10V)

        double vzdalenost=(voltage*4.8428)+30.16;    // Přepočítá napětí na vzdálenost

        write_data(vzdalenost);                      //write data

        if (digitalRead(A10) == HIGH){

            digitalWrite(relay, HIGH);

            //start = false;

            break;

        }

    }

}

void write_data(double vzdalenost)                  //

{

    myFile = SD.open("Laser.csv", FILE_WRITE); //....txt nebo .csv

    if (myFile) {

        myFile.print(Year);

        myFile.print("/");

        myFile.print(Month);

        myFile.print("/");

        myFile.print(Date);

        myFile.print(";");

        myFile.print(Hour);

        myFile.print(":");

        myFile.print(Minute);

        myFile.print(":");

        myFile.print(Second);

        myFile.print(";");

        myFile.print("vzdalenost");

        myFile.print(";");

        myFile.print(vzdalenost);

        myFile.print(";");

```

```
myFile.println(" ");  
myFile.close();  
Serial.print(vzdalenost);  
Serial.print("    ");  
Serial.println("Write file successful!");  
} else {  
    Serial.println("error opening Laser.csv");  
}  
}
```



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.9 ARDUINO NANO DATASHEET

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



element14

---

**EN** - For pricing and availability in your local country please visit one of the below links:

**DE** - Informationen zu Preisen und Verfügbarkeit in Ihrem Land erhalten Sie über die unten aufgeführten Links:

**FR** - Pour connaître les tarifs et la disponibilité dans votre pays, cliquez sur l'un des liens suivants:

[A000005](#)

**EN**

This Datasheet is presented by  
the manufacturer

**DE**

Dieses Datenblatt wird vom  
Hersteller bereitgestellt

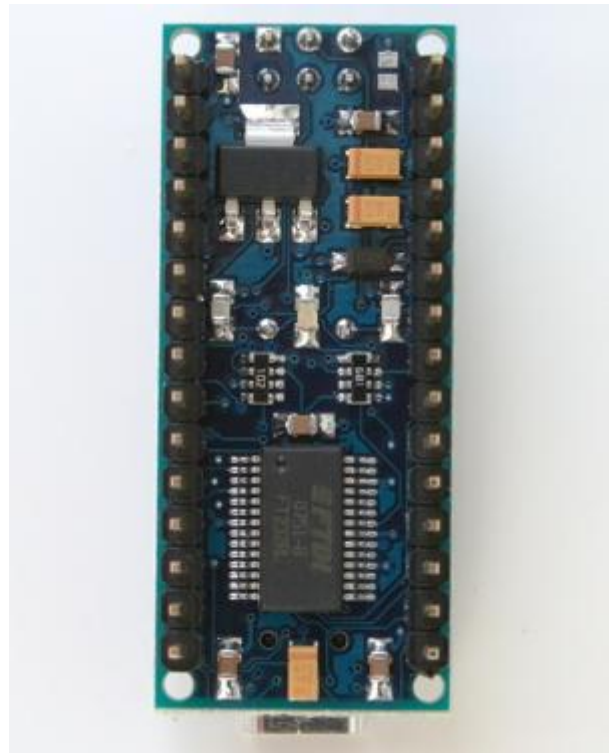
**FR**

Cette fiche technique est  
présentée par le fabricant

# Arduino Nano



*Arduino Nano Front*

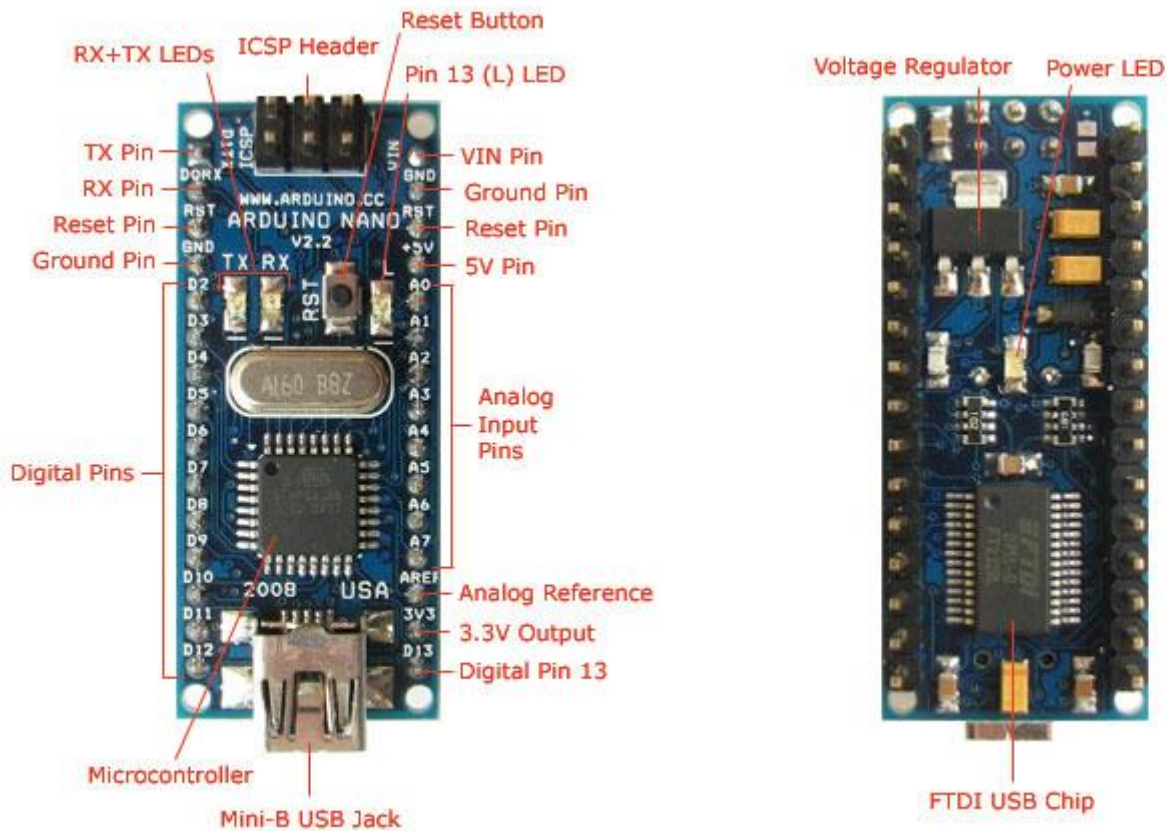


*Arduino Nano Rear*

## Overview

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.0) or ATmega168 (Arduino Nano 2.x). It has more or less the same functionality of the Arduino Duemilanove, but in a different package. It lacks only a DC power jack, and works with a Mini-B USB cable instead of a standard one. The Nano was designed and is being produced by Gravitech.





## Schematic and Design

Arduino Nano 3.0 (ATmega328): [schematic](#), [Eagle files](#).

Arduino Nano 2.3 (ATmega168): [manual](#) (pdf), [Eagle files](#). *Note:* since the free version of Eagle does not handle more than 2 layers, and this version of the Nano is 4 layers, it is published here unrouted, so users can open and use it in the free version of Eagle.

## Specifications:

Microcontroller	Atmel ATmega168 or ATmega328
Operating Voltage (logic level)	5 V
Input Voltage (recommended)	7-12 V
Input Voltage (limits)	6-20 V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	40 mA
Flash Memory	16 KB (ATmega168) or 32 KB (ATmega328) of which 2 KB used by bootloader
SRAM	1 KB (ATmega168) or 2 KB (ATmega328)
EEPROM	512 bytes (ATmega168) or 1 KB (ATmega328)
Clock Speed	16 MHz
Dimensions	0.73" x 1.70"

## Power:

The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

The FTDI FT232RL chip on the Nano is only powered if the board is being powered over USB. As a result, when running on external (non-USB) power, the 3.3V output (which is supplied by the FTDI chip) is not available and the RX and TX LEDs will flicker if digital pins 0 or 1 are high.

## Memory

The ATmega168 has 16 KB of flash memory for storing code (of which 2 KB is used for the bootloader); the ATmega328 has 32 KB, (also with 2 KB used for the bootloader). The ATmega168 has 1 KB of SRAM and 512 bytes of EEPROM (which can be read and written with the [EEPROM library](#)); the ATmega328 has 2 KB of SRAM and 1 KB of EEPROM.

## Input and Output

Each of the 14 digital pins on the Nano can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- ✦ **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- ✦ **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- ✦ **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- ✦ **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- ✦ **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Nano has 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

**I<sup>2</sup>C: 4 (SDA) and 5 (SCL).** Support I<sup>2</sup>C (TWI) communication using the [Wire library](#) (documentation on the Wiring website).

There are a couple of other pins on the board:

**AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).

**Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega168 ports](#).

## Communication

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega168 and ATmega328 provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the [FTDI drivers](#) (included with the Arduino software) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Nano's digital pins.

The ATmega168 and ATmega328 also support I<sup>2</sup>C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I<sup>2</sup>C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega168 or ATmega328 datasheet.

## Programming

The Arduino Nano can be programmed with the Arduino software ([download](#)). Select "Arduino Diecimila, Duemilanove, or Nano w/ ATmega168" or "Arduino Duemilanove or Nano w/ ATmega328" from the **Tools**

> **Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega168 or ATmega328 on the Arduino Nano comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Nano is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the FT232RL is connected to the reset line of the ATmega168 or ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Nano is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Nano. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.



element14

---

**EN** - For pricing and availability in your local country please visit one of the below links:

**DE** - Informationen zu Preisen und Verfügbarkeit in Ihrem Land erhalten Sie über die unten aufgeführten Links:

**FR** - Pour connaître les tarifs et la disponibilité dans votre pays, cliquez sur l'un des liens suivants:

[A000005](#)

**EN**

This Datasheet is presented by  
the manufacturer

**DE**

Dieses Datenblatt wird vom  
Hersteller bereitgestellt

**FR**

Cette fiche technique est  
présentée par le fabricant



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.10 G KOD G39

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUcí PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# 1 G\_KOD G39

Příloha obsahuje popis automatické kalibrace pracovní desky. Tato kalibrace probíhá před každým tiskem.

## Funkce automatické kalibrace (G 39)

Pro automatickou kalibraci byla vytvořena G funkce (G 39). Tato funkce byla obsažena v každém G codu před samotným tiskem. Z důvodu, že software tiskárny posílá příkazy do mezi paměti a příkazy tak nejsou vykonávány „real time“, byla automatická kalibrace rozdělena do skupiny jednotlivých příkazů. Funkce pro automatickou kalibraci je obsažena v řídicí desce tiskárny. Ta posílá jednotlivé příkazy do karty, která ovládá automatickou kalibraci. To probíhá po sériové komunikaci (UART). Mezi příkazy je „uměle“ vytvořena pauza a software čeká na potvrzení, že byl daný příkaz vykonán. Toto potvrzení probíhá pomocí digitálního pinu (změna hodnoty LOW na HIGH). Je to z důvodu urychlení procesu.

## Arduino Nano

Arduino Nano bylo použito jako ovládací karta pro automatickou kalibraci pracovní podložky. Do budoucna může být použito i pro měření teploty pracovní desky a k automatickému ohřevu pracovní desky.

Obsahuje funkce pro ovládání mikro krokových motorků a měření vzdálenosti indukčním čidlem. Veškeré příkazy, které se mají vykonat posílá řídicí deska tiskárny po sériové lince (UART).

Pro servis pracovní desky byly vytvořeny servisní funkce. První funkce obsahuje postupné najetí pracovní desky do spodní polohy, druhá funkce obsahuje postupné vyjetí pracovní desky do horní polohy. Funkce jsou vytvořeny především pro servis pracovní desky z hlediska čištění atd.

## Postup automatické kalibrace:

1. Krok: Po spuštění automatické kalibrace dojde k „Houmingu“ tiskárny. Tiskárna najede do koncové polohy v osách X, Y, Z.
2. Krok: Následně sjede pracovní deska do krajní polohy (spodní).



obr. 1-1 Schéma pohybu

3. Krok: Extrudér s čidlem vzdálenosti najede do středové polohy pracovní desky.
4. Krok: Pracovní deska jede směrem k čidlu vzdálenosti (nahoru), po sepnutí čidla vzdálenosti (pracovní deska se nachází v požadované výšce od trysky) se pohyb pracovní desky zastaví.



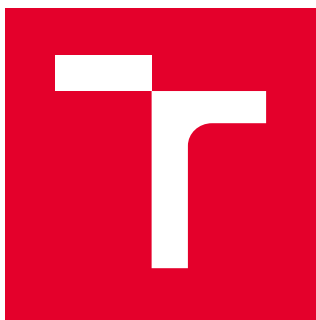
**obr. 1-2** Schéma pohybu

5. Krok: Pracovní deska sjede o daný počet otáček směrem dolů, aby bylo možné zkalibrovat krajní body.
6. Krok: Extrudér s čidlem vzdálenosti najede k prvnímu motoru do kalibračního bodu.
7. Krok: Motor jede směr nahoru, dokud nedojde k sepnutí čidla vzdálenosti a zastavení motoru. Dojde tak k nastavení výšky mezi tryskou a pracovní deskou v daném bodě.



**obr. 1-3** Schéma pohybu

8. Krok: Extrudér s čidlem vzdálenosti najede k druhému motoru do kalibračního bodu.
9. Krok: Motor jede směr nahoru dokud nedojde k sepnutí čidla vzdálenosti a zastavení motoru. Dojde tak k nastavení výšky mezi tryskou a pracovní deskou v daném bodě.
10. Krok: Extrudér s čidlem vzdálenosti najede ke třetímu motoru do kalibračního bodu.
11. Krok: Motor jede směr nahoru dokud nedojde k sepnutí čidla vzdálenosti a zastavení motoru. Dojde tak k nastavení výšky mezi tryskou a pracovní deskou v daném bodě.
12. Krok: Extrudér s čidlem vzdálenosti najede ke čtvrtému motoru do kalibračního bodu.
13. Krok: Motor jede směr nahoru dokud nedojde k sepnutí čidla vzdálenosti a zastavení motoru. Dojde tak k nastavení výšky mezi tryskou a pracovní deskou v daném bodě.
14. Krok: Po dokončení automatické kalibrace dojde k „Homingu“. Tiskárna najede do koncové polohy v osách X, Y, Z. Tímto je kalibrace dokončena a tiskárna je připravena k tisku.



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.11 AUTOMATICKÁ KALIBRACE SOFTWARE

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



# 1 AUTOMATICKÁ KALIBRACE SOFTWARE

Program slouží pro vykonání automatické kalibrace.

Program pro automatickou kalibraci

```
String cmd;
char answer[40];
char c;
int pointer;
bool ok;
inline void gcode_G39() {
    pinMode(11, INPUT);
    SERIAL_ECHOLN("zpracovani kodu g39");
    SERIAL_ECHOLN("home");
    enqueue_and_echo_commands_P(PSTR("G28"));
    SERIAL_ECHOLN("osa Z");
    enqueue_and_echo_commands_P(PSTR("G1 Z30 F1600"));
    SERIAL_ECHOLN("do prostred");
    enqueue_and_echo_commands_P(PSTR("G1 X100 Y150 F1600"));
    enqueue_and_echo_commands_P(PSTR("G40"));
    watchdog_reset();
}
inline void gcode_G40() {
    SERIAL_ECHOLN("...pokracuju G39, ale pod G40 a ted cekam");
    safe_delay(25000);
    SERIAL_ECHOLN("...docekal jsme a jedu...");
    pinMode(11, INPUT);
    SERIAL_ECHOLN("spust spodek - run");
    Serial3.println("run");
    safe_delay(1000);
    watchdog_reset();
    bool beh = true;
```

```

while(beh){
    if(digitalRead(11)==HIGH){
        SERIAL_ECHOLN("Konec fce: run");
        beh = false;
    }
    watchdog_reset();
    delay(100);
}
beh = true;
safe_delay(6000);
enqueue_and_echo_commands_P(PSTR("G1 Z0 F1600"));
enqueue_and_echo_commands_P(PSTR("G41"));
}

inline void gcode_G41() {
    SERIAL_ECHOLN("...pokracuju G39, ale pod G41 a ted cekam");
    safe_delay(15000);
    SERIAL_ECHOLN("...docekal jsme a jedu...");
    pinMode(11, INPUT);
    SERIAL_ECHOLN("spust spodek - run2");
    Serial3.println("run2");
    safe_delay(1000);
    watchdog_reset();
    bool beh = true;
    while(beh){
        if(digitalRead(11)==HIGH){
            SERIAL_ECHOLN("Konec fce: run2");
            beh = false;
        }
        watchdog_reset();
        delay(100);
    }
    beh = true;
}

```

```

enqueue_and_echo_commands_P(PSTR("G1 X180 Y220 F1600"));
enqueue_and_echo_commands_P(PSTR("G43"));
}

inline void gcode_G43() {
    SERIAL_ECHOLN("...pokracuju G39, ale pod G43 (motor1) a ted cekam");
    safe_delay(5000);
    SERIAL_ECHOLN("...docekal jsme a jedu...");
    pinMode(11, INPUT);
    SERIAL_ECHOLN("spust spodek - motor1");
    Serial3.println("motor1");
    safe_delay(1000);
    watchdog_reset();
    bool beh = true;
    while(beh){
        if(digitalRead(11)==HIGH){
            SERIAL_ECHOLN("Konec fce: motor1");
            beh = false;
        }
        watchdog_reset();
        delay(100);
    }
    beh = true;
    enqueue_and_echo_commands_P(PSTR("G1 X180 Y90 F1600"));
    enqueue_and_echo_commands_P(PSTR("G44"));
}

inline void gcode_G44() {
    SERIAL_ECHOLN("...pokracuju G39, ale pod G44 (motor2) a ted cekam");
    safe_delay(5000);
    SERIAL_ECHOLN("...docekal jsme a jedu...");
    pinMode(11, INPUT);

    SERIAL_ECHOLN("spust spodek - motor2");

```

```

Serial3.println("motor2");
safe_delay(1000);
watchdog_reset();
bool beh = true;
while(beh){
    if(digitalRead(11)==HIGH){
        SERIAL_ECHOLN("Konec fce: motor2");
        beh = false;
    }
    watchdog_reset();
    delay(100);
}
beh = true;
enqueue_and_echo_commands_P(PSTR("G1 X40 Y90 F1600"));
enqueue_and_echo_commands_P(PSTR("G45"));
}

inline void gcode_G45() {
    SERIAL_ECHOLN("...pokracuju G39, ale pod G45 (motor3) a ted cekam");
    safe_delay(5000);
    SERIAL_ECHOLN("...docekal jsme a jedu...");
    pinMode(11, INPUT);
    SERIAL_ECHOLN("spust spodek - motor3");
    Serial3.println("motor3");
    safe_delay(1000);
    watchdog_reset();
    bool beh = true;
    while(beh){
        if(digitalRead(11)==HIGH){
            SERIAL_ECHOLN("Konec fce: motor3");
            beh = false;
        }
        watchdog_reset();
    }
}

```

```

        delay(100);
    }
    beh = true;
    enqueue_and_echo_commands_P(PSTR("G1 X40 Y220 F1600"));
    enqueue_and_echo_commands_P(PSTR("G46"));
}

inline void gcode_G46() {
    SERIAL_ECHOLN("...pokracuju G39, ale pod G46 (motor4) a ted cekam");
    safe_delay(5000);
    SERIAL_ECHOLN("...docekal jsme a jedu...");
    pinMode(11, INPUT);
    SERIAL_ECHOLN("spust spodek - motor4");
    Serial3.println("motor4");
    safe_delay(1000);
    watchdog_reset();
    bool beh = true;
    while(beh){
        if(digitalRead(11)==HIGH){
            SERIAL_ECHOLN("Konec fce: motor4");
            beh = false;
        }
        watchdog_reset();
        delay(100);
    }
    beh = true;
    Serial3.println("konec");
    SERIAL_ECHOLN("Koncim a home");
    enqueue_and_echo_commands_P(PSTR("G1 Z30 F1600"));
    enqueue_and_echo_commands_P(PSTR("G28"));
}

```



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.12 AUTOMATICKÁ KALIBRACE ZAPOJENÍ

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

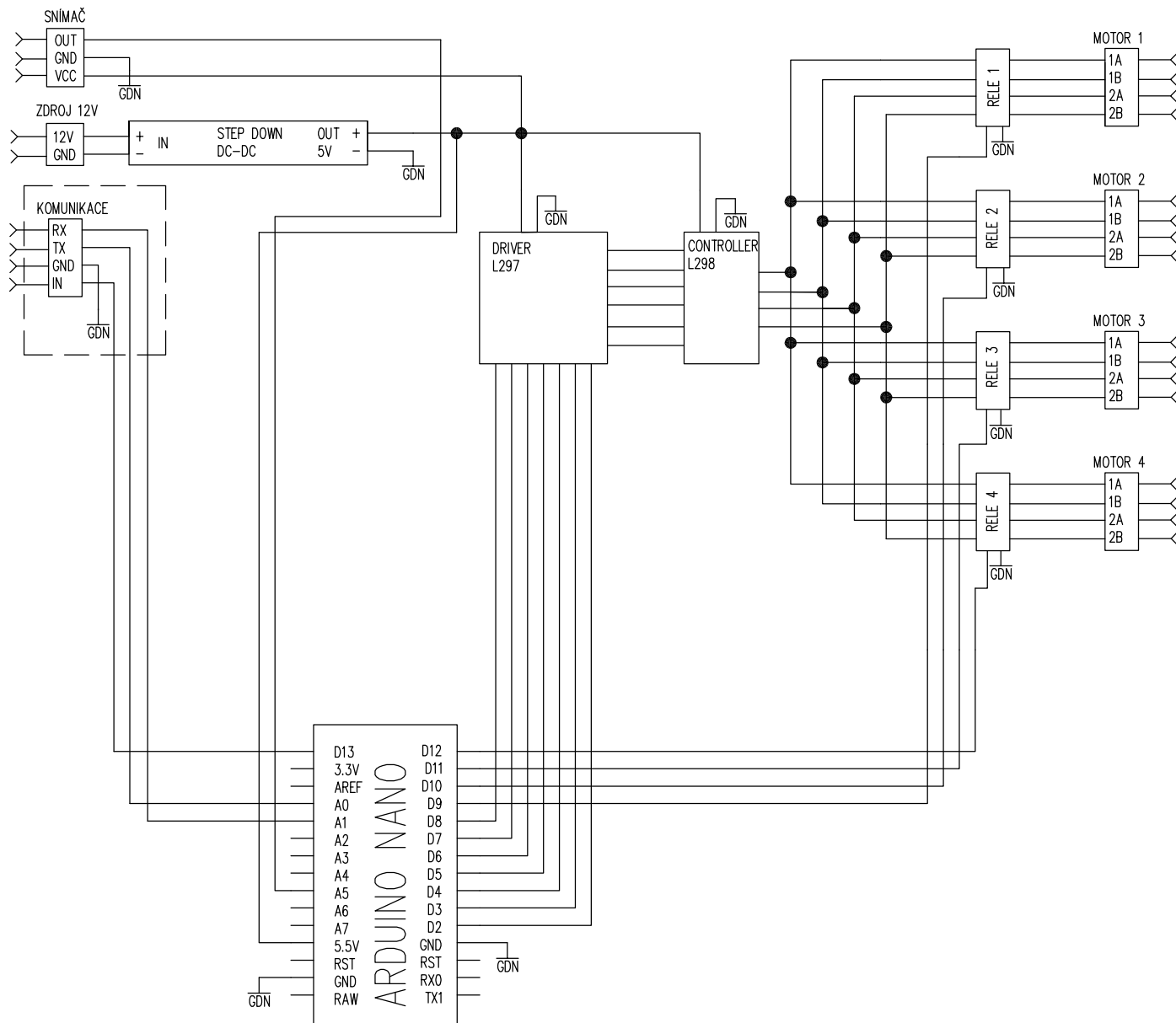
**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE



SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



ZAPOJENÍ  
ZAPOJENÍ KOMUNIKACE

Struktura povrchu:		Hrany:		Měřítko  1:1	Přesnost	PŘESNOST
					Tolerování	TOLEROVÁNÍ
					Promítání	
Materiál _		Polotovar _		Hmotnost _ kg	CHRÁNĚNO PODLE ISO 16016	
  ÚSTAV KONSTRUOVÁNÍ		Druh dokumentu		ELEKTRO_SCHÉMA		
		Kreslil		RAFAJA		
		Schválil		Název		
		Datum vydání		0002		
				ZAPOJENÍ_AUT_KALIB		
				Číslo dokumentu		
				08.05.2019		
				List /		



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.13 USTEPPER S-LITE

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**





Product: **uStepper** *S-lite*

Document revision: 1.1

Author: MGN

Approved by: THO

Approval date: November 27th 2018

# System overview

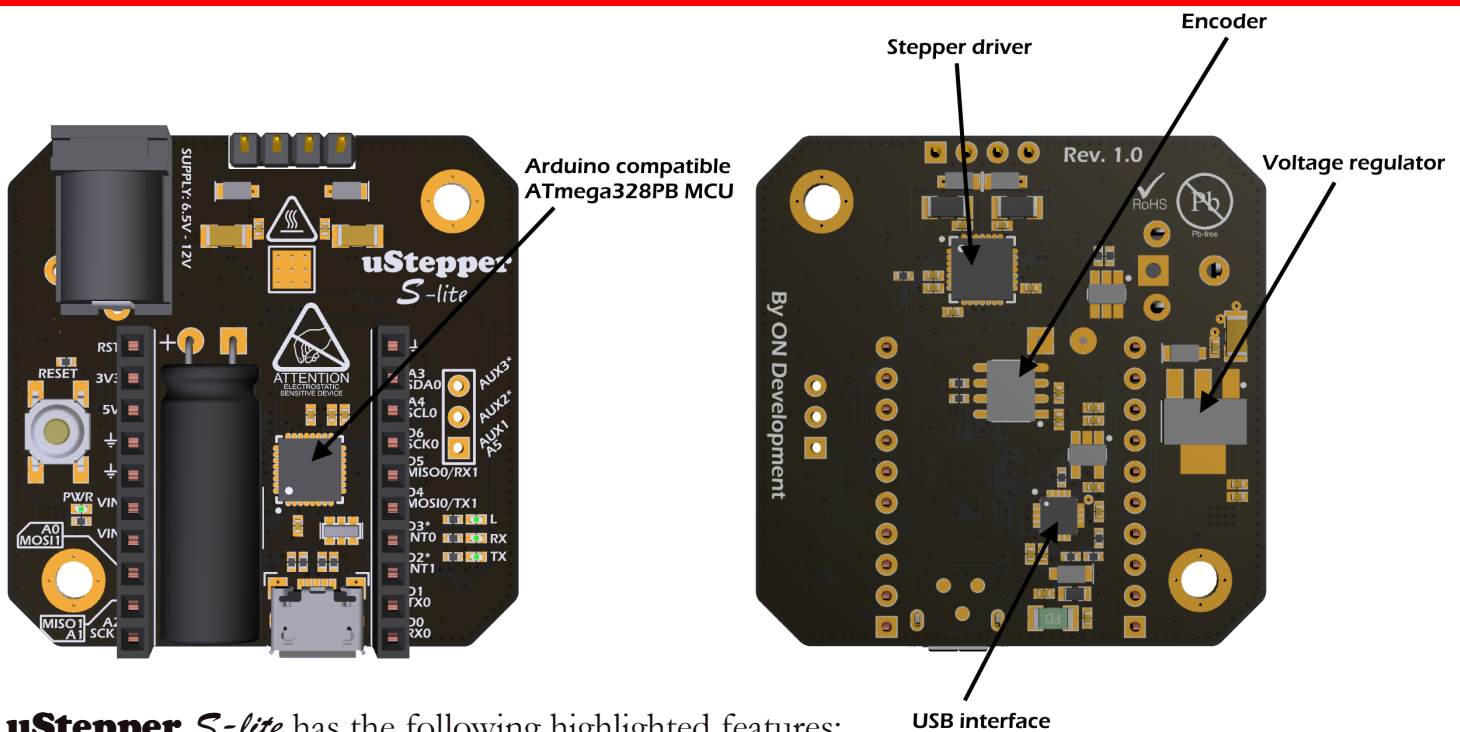
**uStepper S-lite** incorporates super silent stepper drivers, high resolution encoders and Arduino compatible MCU in a super slim design!

- Program uStepper with ease using the Arduino IDE and our extensive Arduino library !
- Make your stepper motor applications extremely compact !
- Make sure that you never miss a step again, thanks to the on-board encoder and control algorithm !



The super slim design of **uStepper S-lite** makes it possible to develop applications using a stepper motor, without the need for long and messy wiring to an external Arduino/stepper shield. Furthermore the 12-bit rotary encoder ensures that the absolute position of the motor shaft can be tracked, enabling the **uStepper S-lite** to detect any loss of steps and act on it !

# uStepper S-lite features



**uStepper S-lite** has the following highlighted features:

- Compact design, fitting on the back of a NEMA 17 size stepper motor (41.8 mm x 41.8 mm)
- Incorporates the Trinamic TMC2208 with dynamic microstepping for very smooth operation
- Driver delivers 2 A current (peak)
- New upgraded 328PB MCU from Microchip
- 12-bit encoder (0.088 degrees resolution)
- Drive current is software controllable
- 15 I/O
- 6 Analog, 7 PWM
- 2 x SPI , 2 x UART, 1 x I2C
- Supply voltage 6.5 - 12 V
- Reverse polarity protection

Even though **uStepper S-lite** features reverse polarity protection, it is always advisable to check the polarity before supplying power to the board in order to prevent damaging the board.

# uStepper *S-lite* features

## MCU

**uStepper *S-lite*** uses the successor of the well tested and widely used ATmega328P - the ATmega328PB. Why? Because it does the job and does it good! The new upgraded ATmega328PB has been upgraded on multiple points including double USART, double I2C, double SPI and upgraded timers !

## Stepper driver chip

The TMC2208 features a whopping 1/256th microstepping interpolation algorithm, which means that it takes in 1/16th microstepping pulses, as send out by most 3D printer controllers for example, and dividing each of these into 16 substeps ! This means that the rotor inside the motor travels much shorter distances between the steps, and hereby significantly reduces the resonances generated by stepper motors. This in turn ensures that you gets the smoothest possible operation while keeping the noise to an absolute minimum !

## Encoder

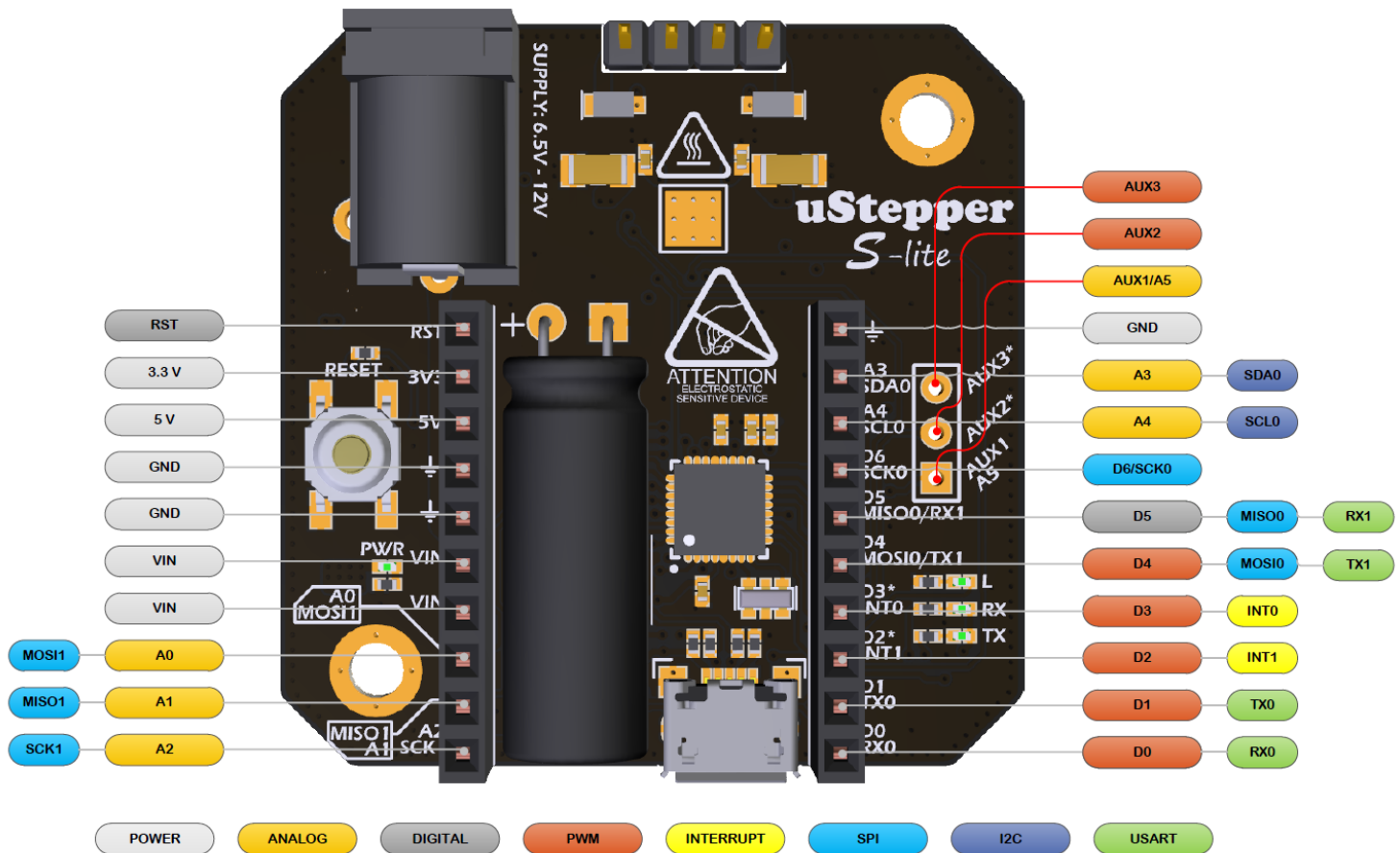
The **uStepper *S-lite*** uses the AS5600 encoder from AMS, delivering 12 bits resolution and delivers great performance for the money.

## Software Features

**uStepper *S-lite*** uses the code base of the original **uStepper**. This will give you a lot of functionality, and some of the key ones are:

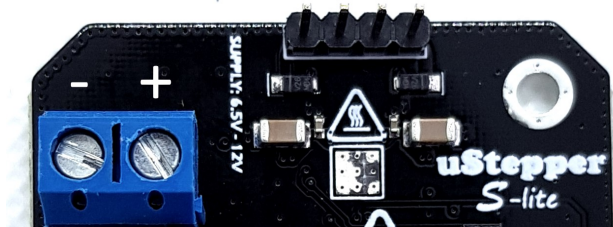
- Closed loop position control
- Drop-in capability
- Simple move/stop commands
- Stall detection feedback
- Encoder diagnostics
- Multi-turn absolute position feedback (almost infinite revolutions)

# uStepper S-lite pin mapping



As it can be seen from the PIN mapping above, **uStepper S-lite** offers a vast amount of GPIO's from it's ATmega328PB MCU. This makes the uStepper capable of interacting with various sensors, actuators, communication modules etc. - Besides functioning as a stepper driver with feedback capability.

## Polarity Screw Terminal



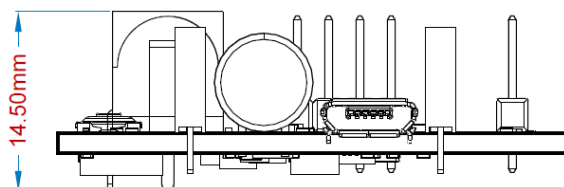
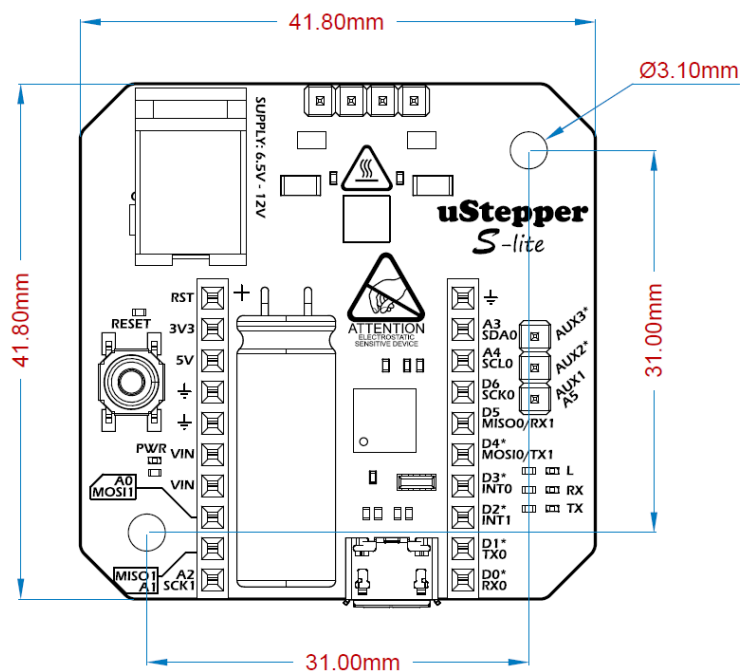
# uStepper *S-lite* characteristics

## Electrical characteristics

I/O Voltage	5 V
Input Voltage (recommended)	6.5 - 12 V
DC Current per I/O Pin (max)	40 mA (max 200 mA total I/O draw)
DC Current for 5 V Pin (max)	200 mA
DC Current for 3.3 V Pin (max)	50 mA
Stepper drive current	Up to 2A (peak)

## Physical characteristics

Dimensions	41.8 x 41.8 x 14.5 mm
Weight	~13g



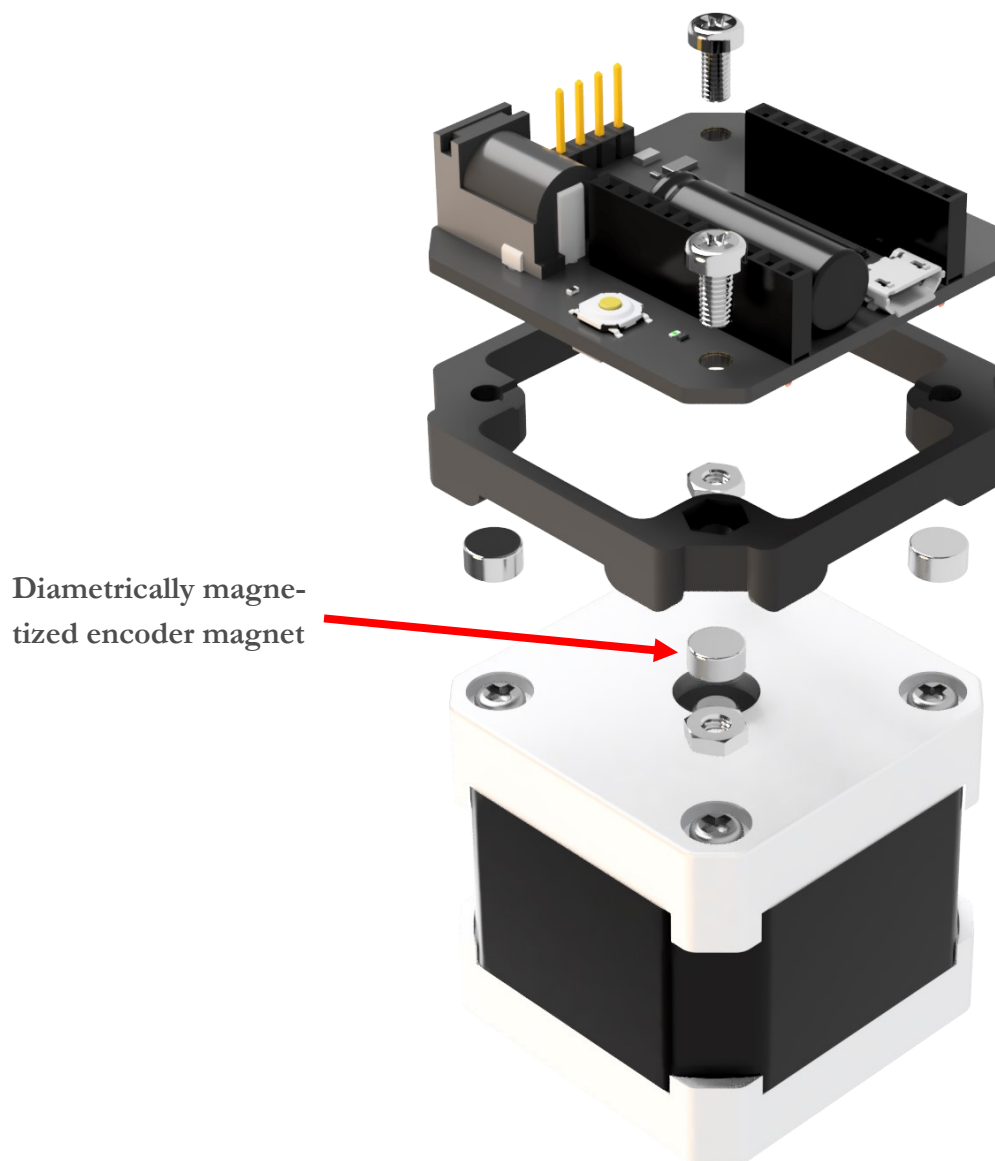


# uStepper *S-lite* mounting

**uStepper *S-lite*** comes with a bracket allowing you to easily mount it on the back of a NEMA 17 stepper motor ! And here is how you do it:

## Magnet mount:

1. Place encoder magnet on motor shaft
2. Insert magnets into bracket
3. Secure **uStepper *S-lite*** to the bracket with screws and nuts
4. Click on the bracket





## uStepper *S-lite* FAQ

For information on how to setup uStepper in the Arduino IDE, visit our GitHub repository at [www.github.com/uStepper](https://www.github.com/uStepper) or have a look at our videos on YouTube - search for uStepper.

More information/documentation on the uStepper Arduino library is also found in the readme section on our GitHub repository.

**Q:** I can't program uStepper from the Arduino IDE, what is wrong?

**A:** Three most common issues:

1. If you use the Windows store app "Arduino IDE" you might experience problems. Download the executable or portable version of Arduino IDE from [www.arduino.cc](http://www.arduino.cc)
2. Install hardware support and the **uStepper *S-lite*** library
3. Remember to choose **uStepper *S-lite*** as the board you program to

**Q:** When I run uStepper closed loop PID or Drop-in the motor spins up and runs fast in one direction, what is wrong?

**A:** Two most common issues:

1. Check that the encoder magnet is placed correctly on the motor shaft
2. Check that your PID parameters in the Arduino code are sane. A good starting point is to set P and I to 0.5 and D to 0.

For more FAQ visit [www.ustepper.com](http://www.ustepper.com) !

## **uStepper *S-lite* applications**

Because of low cost and reliability stepper motors are favored over servo drives in a variety of applications, as for example robots, CNC machines, 3D-printers etc. With added feedback uStepper can add a new dimension to stepper operation utilizing closed loop control or correction of missed steps - making the stepper even more appealing than a more costly servo drive. With the numerous features and the vast amount of I/O's (including various busses), uStepper is the choice for DIY hobbyists when in need of a reliable, precise and compact actuator for almost any application.

Almost unlimited applications and ease of programming using the Arduino IDE also makes uStepper well suited for educational purposes !

***"After all, learning just gets a bit more fun when there's movement involved!"***

# Disclaimer

## 1 Disclaimers and Limitation of Liability

1.1 uStepper ApS and/or ON Development IVS (or any individuals affiliated with either of the two companies) can not be held responsible for any damage inflicted upon mounting or interfacing with the uStepper board. This also includes damage to stepper motor (both electrical and mechanical) or any other 3rd party hardware connected to uStepper. Most stepper motor cases are made of aluminum, and care must be taken when preparing the mountings for uStepper.

1.2 By using the uStepper products (including, but not limited to, hardware and software) you acknowledge that uStepper ApS and/or ON Development IVS (or any individuals affiliated with either of the two companies) can not be held responsible for any personal injuries and/or damage to any 3rd party hardware that may occur when using the uStepper products.

1.3 To the extent permitted by law, uStepper ApS and/or ON Development IVS will not be liable for any indirect or consequential loss or damage, of any kind, (including without limitation loss of business, opportunity, data, profits) arising out of or in connection with the use of any products (including, but not limited to, hardware and software), developed, produced or sold by uStepper ApS and/or ON Development IVS (or any individuals affiliated with either of the two companies).

1.4 Nothing in these Terms and Conditions shall be construed so as to hold uStepper ApS and/or ON Development IVS liable for death or personal injury as a result of the negligence of uStepper ApS and/or ON Development IVS or that of its employees or agents.

## 2 Indemnity

2.1 You agree to indemnify and hold uStepper ApS and/or ON Development IVS and its employees and agents harmless from and against all liabilities, legal fees, damages, losses, costs and other expenses in relation to any claims or actions brought against uStepper ApS and/or ON Development IVS arising out of any breach by you of these Terms and Conditions or other liabilities arising out of your use of this Website.

2.2 You agree to indemnify and hold uStepper ApS and/or ON Development IVS and its employees and agents harmless from and against all liabilities, legal fees, damages, losses, costs and other expenses in relation to any claims or actions brought against uStepper ApS and/or ON Development IVS arising out of any breach by you of these Terms and Conditions or other liabilities arising out of your use of products (including, but not limited to, hardware and software) developed, produced or sold by uStepper ApS and/or ON Development IVS (or any individuals affiliated with either of the two companies).

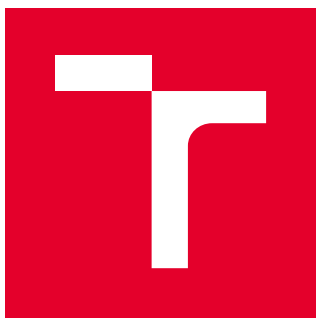
## 3 Severance

If any of these Terms and Conditions should be determined to be invalid, illegal or unenforceable for any reason by any court of competent jurisdiction then such Term or Condition shall be severed and the remaining Terms and Conditions shall survive and remain in full force and effect and continue to be binding and enforceable.

## Contact

uStepper ApS  
sales@ustepper.com

[www.ustepper.com](http://www.ustepper.com)



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.14 ENKODÉRY SOFTWARE

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUcí PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# 1 ENKODÉRY SOFTWARE

Program slouží pro monitorování polohy trysky pomocí enkodéru.

Program monitorovacího modulu

```
#include <uStepperSLite.h>
```

```
uStepperSLite stepper;
```

```
void setup(void)
```

```
{
```

```
    stepper.setup(DROPIN,3200.0,1.0,0.02,0.006,true);
```

```
}
```

```
void loop(void)
```

```
{
```

```
}
```



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.15 ARDUINO GY-521

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



## MPU-6000/MPU-6050 Product Specification

Document Number: PS-MPU-6000A-00  
Revision: 3.3  
Release Date: 5/16/2012

### Primary Differences between MPU-6000 and MPU-6050

Part / Item	MPU-6000	MPU-6050
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	n/a	1.71V to VDD
Serial Interfaces Supported	I <sup>2</sup> C, SPI	I <sup>2</sup> C
Pin 8	/CS	VLOGIC
Pin 9	AD0/SDO	AD0
Pin 23	SCL/SCLK	SCL
Pin 24	SDA/SDI	SDA



## 5 Features

### 5.1 Gyroscope Features

The triple-axis MEMS gyroscope in the MPU-60X0 includes a wide range of features:

- Digital-output X-, Y-, and Z-Axis angular rate sensors (gyroscopes) with a user-programmable full-scale range of  $\pm 250$ ,  $\pm 500$ ,  $\pm 1000$ , and  $\pm 2000^\circ/\text{sec}$
- External sync signal connected to the FSYNC pin supports image, video and GPS synchronization
- Integrated 16-bit ADCs enable simultaneous sampling of gyros
- Enhanced bias and sensitivity temperature stability reduces the need for user calibration
- Improved low-frequency noise performance
- Digitally-programmable low-pass filter
- Gyroscope operating current: 3.6mA
- Standby current: 5 $\mu$ A
- Factory calibrated sensitivity scale factor
- User self-test

### 5.2 Accelerometer Features

The triple-axis MEMS accelerometer in MPU-60X0 includes a wide range of features:

- Digital-output triple-axis accelerometer with a programmable full scale range of  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$  and  $\pm 16g$
- Integrated 16-bit ADCs enable simultaneous sampling of accelerometers while requiring no external multiplexer
- Accelerometer normal operating current: 500 $\mu$ A
- Low power accelerometer mode current: 10 $\mu$ A at 1.25Hz, 20 $\mu$ A at 5Hz, 60 $\mu$ A at 20Hz, 110 $\mu$ A at 40Hz
- Orientation detection and signaling
- Tap detection
- User-programmable interrupts
- High-G interrupt
- User self-test

### 5.3 Additional Features

The MPU-60X0 includes the following additional features:

- 9-Axis MotionFusion by the on-chip Digital Motion Processor (DMP)
- Auxiliary master I<sup>2</sup>C bus for reading data from external sensors (e.g., magnetometer)
- 3.9mA operating current when all 6 motion sensing axes and the DMP are enabled
- VDD supply voltage range of 2.375V-3.46V
- Flexible VLOGIC reference voltage supports multiple I<sup>2</sup>C interface voltages (MPU-6050 only)
- Smallest and thinnest QFN package for portable devices: 4x4x0.9mm
- Minimal cross-axis sensitivity between the accelerometer and gyroscope axes
- 1024 byte FIFO buffer reduces power consumption by allowing host processor to read the data in bursts and then go into a low-power mode as the MPU collects more data
- Digital-output temperature sensor
- User-programmable digital filters for gyroscope, accelerometer, and temp sensor
- 10,000 g shock tolerant
- 400kHz Fast Mode I<sup>2</sup>C for communicating with all registers
- 1MHz SPI serial interface for communicating with all registers (MPU-6000 only)
- 20MHz SPI serial interface for reading sensor and interrupt registers (MPU-6000 only)





## MPU-6000/MPU-6050 Product Specification

Document Number: PS-MPU-6000A-00  
Revision: 3.3  
Release Date: 5/16/2012

### 6.2 Accelerometer Specifications

VDD = 2.375V-3.46V, VLOGIC (MPU-6050 only) = 1.8V±5% or VDD, T<sub>A</sub> = 25°C

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
<b>ACCELEROMETER SENSITIVITY</b>						
Full-Scale Range	AFS_SEL=0		±2		g	
	AFS_SEL=1		±4		g	
	AFS_SEL=2		±8		g	
	AFS_SEL=3		±16		g	
ADC Word Length	Output in two's complement format		16		bits	
Sensitivity Scale Factor	AFS_SEL=0		16,384		LSB/g	
	AFS_SEL=1		8,192		LSB/g	
	AFS_SEL=2		4,096		LSB/g	
	AFS_SEL=3		2,048		LSB/g	
Initial Calibration Tolerance			±3		%	
Sensitivity Change vs. Temperature	AFS_SEL=0, -40°C to +85°C		±0.02		%/°C	
Nonlinearity	Best Fit Straight Line		0.5		%	
Cross-Axis Sensitivity			±2		%	
<b>ZERO-G OUTPUT</b>						
Initial Calibration Tolerance	X and Y axes		±50		mg	1
	Z axis		±80		mg	
Zero-G Level Change vs. Temperature	X and Y axes, 0°C to +70°C		±35			
	Z axis, 0°C to +70°C		±60		mg	
<b>SELF TEST RESPONSE</b>						
Relative	Change from factory trim	-14		14	%	2
<b>NOISE PERFORMANCE</b>						
Power Spectral Density	@10Hz, AFS_SEL=0 & ODR=1kHz		400		μg/√Hz	
<b>LOW PASS FILTER RESPONSE</b>						
	Programmable Range	5		260	Hz	
<b>OUTPUT DATA RATE</b>						
	Programmable Range	4		1,000	Hz	
<b>INTELLIGENCE FUNCTION INCREMENT</b>			32		mg/LSB	

1. Typical zero-g initial calibration tolerance value after MSL3 preconditioning
2. Please refer to the following document for further information on Self-Test: *MPU-6000/MPU-6050 Register Map and Descriptions*



## MPU-6000/MPU-6050 Product Specification

Document Number: PS-MPU-6000A-00  
Revision: 3.3  
Release Date: 5/16/2012

### 6.3 Electrical and Other Common Specifications

VDD = 2.375V-3.46V, VLOGIC (MPU-6050 only) = 1.8V±5% or VDD, T<sub>A</sub> = 25°C

PARAMETER	CONDITIONS	MIN	TYP	MAX	Units	Notes
<b>TEMPERATURE SENSOR</b>						
Range			-40 to +85		°C	
Sensitivity	Untrimmed		340		LSB/°C	
Temperature Offset	35°C		-521		LSB	
Linearity	Best fit straight line (-40°C to +85°C)		±1		°C	
<b>VDD POWER SUPPLY</b>						
Operating Voltages		2.375		3.46	V	
Normal Operating Current	Gyroscope + Accelerometer + DMP		3.9		mA	
	Gyroscope + Accelerometer (DMP disabled)		3.8		mA	
	Gyroscope + DMP (Accelerometer disabled)		3.7		mA	
	Gyroscope only (DMP & Accelerometer disabled)		3.6		mA	
	Accelerometer only (DMP & Gyroscope disabled)		500		µA	
Accelerometer Low Power Mode Current	1.25 Hz update rate		10		µA	
	5 Hz update rate		20		µA	
	20 Hz update rate		70		µA	
	40 Hz update rate		140		µA	
Full-Chip Idle Mode Supply Current			5		µA	
Power Supply Ramp Rate	Monotonic ramp. Ramp rate is 10% to 90% of the final value			100	ms	
<b>VLOGIC REFERENCE VOLTAGE</b>						
Voltage Range	MPU-6050 only	1.71		VDD	V	
Power Supply Ramp Rate	VLOGIC must be ≤ VDD at all times Monotonic ramp. Ramp rate is 10% to 90% of the final value			3	ms	
Normal Operating Current			100		µA	
<b>TEMPERATURE RANGE</b>						
Specified Temperature Range	Performance parameters are not applicable beyond Specified Temperature Range	-40		+85	°C	



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.16 BRUEL KJAER 4393 DATASHEET

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUcí PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

## Piezoelectric Charge Accelerometer Types 4393 and 4393-V

### Uses

- High-level, high-frequency measurements
- Vibration testing and analysis
- Shock measurements
- Measurements on delicate structures, in confined spaces or in high-temperature environments

### Features

- Small size
- Low weight
- Very high resonance frequency

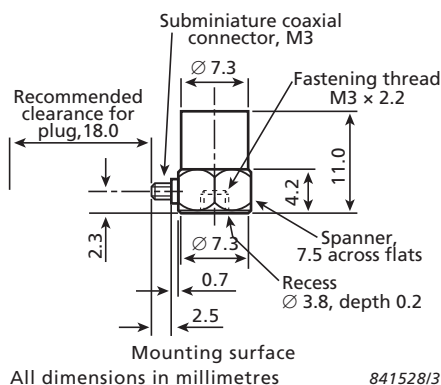


040128

### Description

Type 4393 is a miniature DeltaShear™ Unigain\* accelerometer with a low weight and high resonance frequency. It has an M3 side connector and an M3 threaded hole for mounting. Type 4393-V† has the same specifications and long-term stability as Type 4393, but it has a relaxed sensitivity tolerance.

**Fig. 1** Dimensions of Type 4393



### Characteristics

This piezoelectric accelerometer may be treated as a charge source. Its sensitivity is expressed in terms of charge per unit acceleration ( $\text{pC/ms}^{-2}$ ,  $\text{pC/g}$ ).

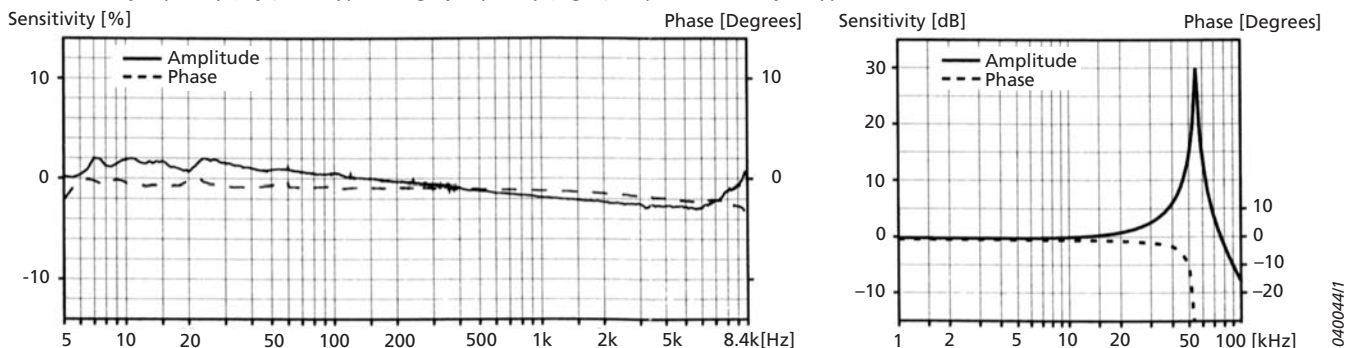
The DeltaShear design consists of three piezoelectric elements and three seismic masses arranged in a triangular configuration around a triangular centre post. They are held in place by a clamping ring that isolates the configuration from the base. The ring also prestresses the piezoelectric elements to give a high degree of linearity. This design provides a high sensitivity-to-mass ratio, a relatively high resonance frequency and high isolation from base strains and temperature transients.

The piezoelectric element used in Type 4393 is PZ 23, and the housing material is titanium.

### Calibration

The sensitivity is calibrated to a convenient value such as 1, 3.16 or  $31.6 \text{ pC/ms}^{-2}$  for Unigain accelerometers. The sensitivity given in the calibration chart has been measured at 159.2 Hz with 95% confidence level, using the coverage factor  $k = 2$ .

**Fig. 2** Individual frequency (left) and typical high-frequency (right) response curves for Type 4393



\* Unigain: The individual measured sensitivity is within  $\pm 2\%$  of the specified sensitivity

† V-type: The individual measured sensitivity is within  $\pm 15\%$  of the specified sensitivity





# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.17 VIBRACE SOFTWARE

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**

# 1 VIBRACE SOFTWARE

Program slouží pro měření vibrací a následné vyhodnocování dat.

Program pro monitorovací modul

```
#include<Wire.h>

const int MPU_addr=0x68;
int16_t AcX,AcY,AcZ;
double vibrace;
double vibraceAz;
double vibraceAzg;
int i;
int j;
boolean start = true;
void setup(){
  Wire.begin();
  Wire.beginTransmission(MPU_addr);
  Wire.write(0x6B);
  Wire.write(0);
  Wire.endTransmission(true);
  Serial.begin(250000);
  pinMode(22,OUTPUT);
  pinMode(23,OUTPUT);
  Serial.println("ready");
}
void loop(){
  if (start == true){
    Wire.beginTransmission(MPU_addr);
    Wire.write(0x3B);
    Wire.endTransmission(false);
    Wire.requestFrom(MPU_addr,14,true);
    AcZ=Wire.read()<<8|Wire.read();
```

```

vibrace = AcZ;
vibraceAz= vibrace-3698;
vibraceAzg= vibraceAz/300;           // pouzitelna hodnota ms-2
if (vibraceAzg > 10 ){
j=1;
delay(100);
for(i=0; i < 6000; i++){
    //nove mereni
    Wire.beginTransmission(MPU_addr);
    Wire.write(0x3B);
    Wire.endTransmission(false);
    Wire.requestFrom(MPU_addr,14,true);
    AcZ=Wire.read()<<8|Wire.read();
    vibrace = AcZ;
    vibraceAz= vibrace-3698;
    vibraceAzg= vibraceAz/300;       // pouzitelna hodnota ms-2
    if (vibraceAzg > 10 ){
        delay(250); //500
        j++;
        if (j>10){
            start = false;        //
            Serial.println("konec");
            digitalWrite(23,HIGH);
            delay(10000);
            break;
        }
    }
    delay(1);
}
}

//Overení vzorkovací frequence
digitalWrite(22,HIGH);

```



```
digitalWrite(22,LOW);  
}  
}
```

Program pro zastavení tisku

```
if (start == true){  
  pinMode(6,INPUT);  
  if(digitalRead(6)==HIGH){  
    i++;  
    if(i>=5){  
      for(int f=0;f<6;f++){  
        enqueue_and_echo_commands_P(PSTR("M84 X Y Z E"));  
        SERIAL_ECHOLN("zastavení.....");  
        card.stopSDPrint();  
        safe_delay(1000);  
        watchdog_reset();  
      }  
      safe_delay(1000);  
      enqueue_and_echo_commands_P(PSTR("G72"));  
      watchdog_reset();  
      safe_delay(1000);  
      start=false;  
      watchdog_reset();  
    }  
  }  
}
```



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV KONSTRUOVÁNÍ

INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

## PŘÍLOHA Č.18 ZAPOJENÍ MONITOROVÁNÍ POMOCÍ VIBRACÍ

### MONITOROVÁNÍ PROCESU FDM TISKU

MONITORING OF FDM PRINTING PROCESS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

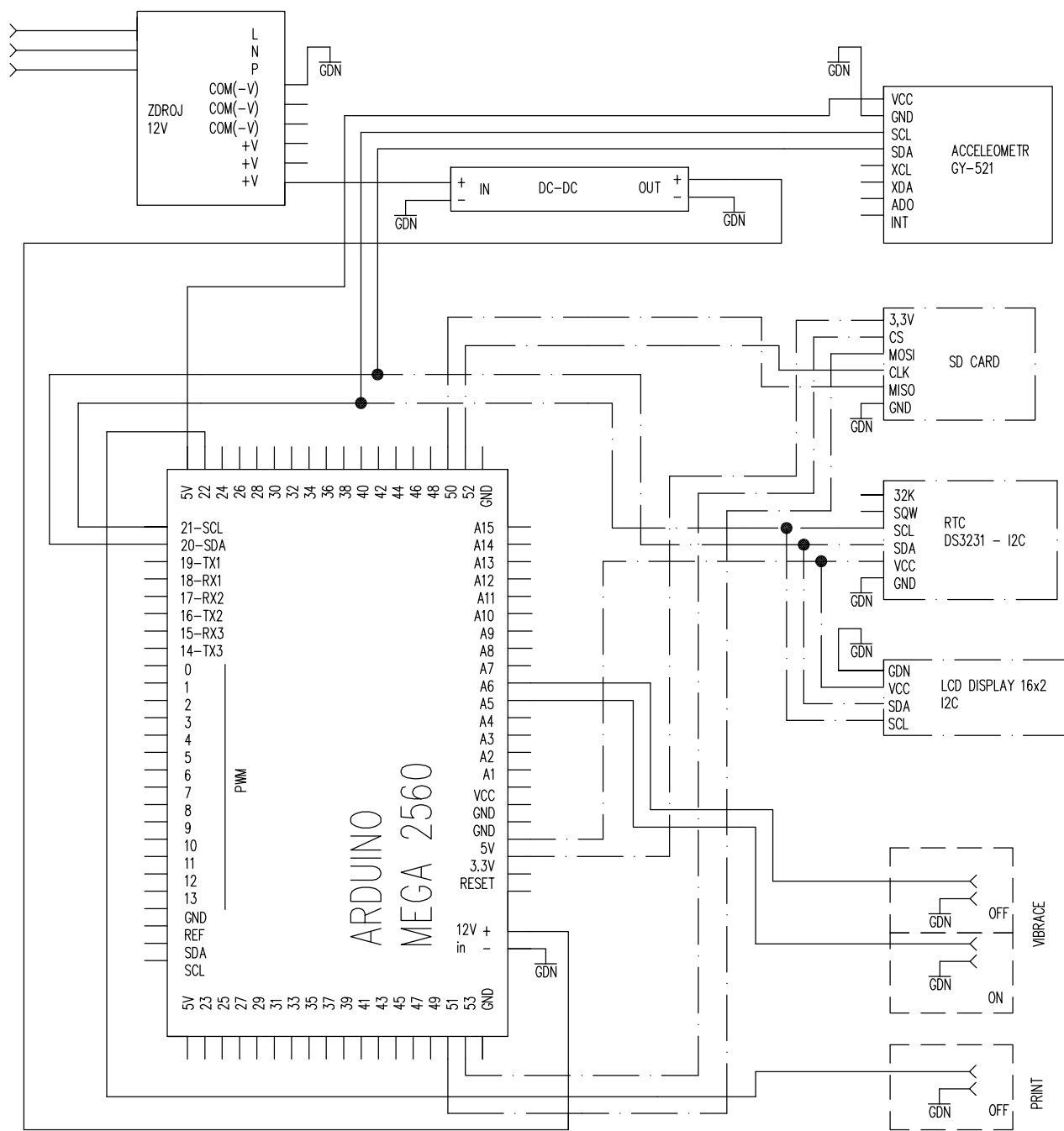
**Bc. Hynek Rafaja**

### VEDOUCÍ PRÁCE



SUPERVISOR

**Ing. Aneta Zatočilová, Ph.D.**

**BRNO 2019**



ZAPOJENÍ VIBRAČÍ  
 ZAPOJENÍ KOMUNIKACE  
 ZAPOJENÍ MODULŮ (SD..)–NEPOUŽITO

Struktura povrchu:		Hrany:		Měřítko  1:1	Přesnost <b>PŘESNOST</b>
					Tolerování <b>TOLEROVÁNÍ</b>
					Promítání 
Materiál _		Polotovar _		Hmotnost _ kg	CHRÁNĚNO PODLE ISO 16016
  ÚSTAV KONSTRUOVÁNÍ		Druh dokumentu <b>ELEKTRO_SCHÉMA</b>		Název  <b>ZAPOJENÍ_VIBRACE</b>	
		Kreslil <b>RAFAJA</b>			
		Schválil		Číslo dokumentu  <b>0004</b>	
		Datum vydání <b>08.05.2019</b>			
List /					